



**General Services Administration
(GSA)**

Federal Acquisition Service (FAS)

**Systems Development Life Cycle (SDLC)
Guidance Document,
Version 2.1**

February 2012

TABLE OF CONTENTS

REVISION HISTORY	VI
EXECUTIVE SUMMARY	1
1.0 INTRODUCTION	1
1.1 Background	1
1.2 Purpose	1
1.3 Scope	1
1.4 Assumptions	1
1.5 SDLC Overview	2
1.5.1 SDLC Stages	2
1.5.1.1 Concept Approval Stage	3
1.5.1.2 Project Planning Stage	3
1.5.1.3 Requirements Analysis Stage	4
1.5.1.4 Design Stage	4
1.5.1.5 Development/Integration and Test Stage	4
1.5.1.6 Implementation Stage	5
1.5.1.7 Operations and Maintenance Stage	5
1.5.1.8 Disposition Stage	5
1.6 Documentation Requirements	5
1.7 Key Activities Related to System Development	7
1.7.1 Strategic Planning	7
1.7.2 Capital Planning and Investment Control (CPIC)	7
1.7.3 Systems Security	7
1.7.4 Enterprise Architecture	7
1.7.5 Performance Management	8
1.7.6 Business Process Re-engineering	8
2.0 CONCEPT APPROVAL STAGE.....	9
2.1 Overview	9
2.1.1 Objectives	9
2.1.2 Inputs	9
2.1.3 High Level Activities	9
2.1.4 Work Products	9
2.2 Tasks and Activities.....	9
2.2.1 Identify the Opportunity to Improve Business Functions.....	9
2.2.2 Identify a Program Sponsor.....	9
2.2.3 Create Documentation and Deliverables.....	10
2.2.3.1 Prepare Draft Business Case	10
2.2.3.2 Analyze Fit with the Enterprise Architecture	10
2.2.4 Hold Stage Review Activity.....	10
3.0 PLANNING STAGE	11
3.1 Overview	11
3.1.1 Objectives.....	11
3.1.2 Inputs	11
3.1.3 High Level Activities.....	11
3.1.4 Work Products	11
3.2 Tasks and Activities.....	12

3.2.1	Form (or appoint) a Project Organization	12
3.2.2	Study and Analyze the Business Need.....	12
3.2.3	Form the Project Acquisition Strategy.....	12
3.2.4	Create Internal Processes	13
3.2.5	Determine Security Categorization.....	13
3.2.6	Analyze Project Schedule and Cost.....	13
3.2.7	Establish Agreements with Stakeholders	13
3.2.8	Plan the Solicitation, Selection and Award.....	14
3.2.9	Create Documentation and Deliverables.....	14
3.2.9.1	Draft Business Case (DBC)	14
3.2.9.2	Security Risk Assessment Report.....	15
3.2.9.3	System Security Plan	15
3.2.9.4	Acquisition Plan.....	15
3.2.9.5	Project Management Plan	15
3.2.9.6	Concept of Operations	16
3.2.9.7	FAS Earned Value Management (EVM) Package.....	16
3.2.10	Hold Stage Review Activity.....	16
4.0	REQUIREMENTS ANALYSIS STAGE	17
4.1	Overview	17
4.1.1	Objectives.....	17
4.1.2	Inputs	17
4.1.3	High Level Activities.....	17
4.1.4	Work Products	17
4.2	Tasks and Activities.....	17
4.2.1	Define Requirements.....	18
4.2.1.1	Define User Interface Requirements	18
4.2.1.2	Define Security Requirements	18
4.2.1.3	Define FOIA/Privacy Act Requirements	18
4.2.1.4	Define Data Requirements	19
4.2.1.5	Define Other Requirements	19
4.2.2	Create Documentation and Deliverables.....	19
4.2.2.1	Develop Requirements Traceability Matrix	19
4.2.2.2	Develop Functional Requirements Document.....	19
4.2.2.3	Develop an Interface Control Document.....	20
4.2.2.4	Develop Test Criteria and Plans.....	20
4.2.3	Conduct System Requirements Review	20
4.2.4	Establish Functional Baseline	20
4.2.5	Hold Stage Review Activity.....	20
5.0	DESIGN STAGE	22
5.1	Overview	22
5.1.1	Objectives.....	22
5.1.2	Inputs	22
5.1.3	High Level Activities.....	22
5.1.4	Work Products	22
5.2	Tasks and Activities.....	22
5.2.1	Determine System Structure	23
5.2.2	Define the Application & Development Environments.....	23
5.2.3	Design the System.....	23
5.2.3.1	Design User Interface	23
5.2.3.2	Design System Interfaces.....	24
5.2.3.3	Design System Security Controls.....	24
5.2.3.4	Build Logical Data Model.....	24

5.2.3.5	Define/Confirm System Architecture.....	24
5.2.3.6	Design Physical Model and Database Structure.....	24
5.2.3.7	Update the Requirements Traceability Matrix	25
5.2.4	Conduct Preliminary Design Review.....	25
5.2.5	Create Documentation and Deliverables.....	25
5.2.5.1	Develop Conversion/Migration/Transition Strategies	25
5.2.5.2	Develop System Design Document	25
5.2.5.3	Develop Implementation Plan	25
5.2.5.4	Develop Maintenance Manual	25
5.2.5.5	Develop Operations Manual/System Administration Manual	26
5.2.5.6	Design User Training	26
5.2.5.7	Develop User Manual.....	26
5.2.5.8	Develop IT Contingency Plan	26
5.2.6	Conduct Critical Design Review.....	26
5.2.7	Hold Stage Review Activity.....	27
6.0	DEVELOPMENT/INTEGRATION & TEST STAGE	28
6.1	Overview	28
6.1.1	Objectives.....	28
6.1.2	Inputs	29
6.1.3	High Level Activities.....	29
6.1.4	Work Products	29
6.2	Tasks and Activities.....	29
6.2.1	Establish the Development and Test Environments	29
6.2.2	Perform Development.....	29
6.2.2.1	Develop the Software	29
6.2.2.2	Develop the Database	29
6.2.3	Conduct Unit Test	30
6.2.4	Integrate Software.....	30
6.2.5	Install Software	30
6.2.6	Develop/Integrate System Security Controls	30
6.2.7	Establish the Test Environments	30
6.2.8	Conduct Integration Test.....	30
6.2.9	Hold Test Readiness Reviews	30
6.2.10	Conduct System Testing.....	31
6.2.11	Conduct Security Testing.....	31
6.2.12	Conduct Acceptance Testing	31
6.2.13	Create Documentation and Deliverables.....	32
6.2.13.1	Prepare Software Development Folder.....	32
6.2.13.2	Deliver System Software	32
6.2.13.3	Deliver Test Files/Data	32
6.2.13.4	Deliver Test Analysis Report	32
6.2.13.5	Deliver Problem Report	32
6.2.13.6	Deliver User Acceptance Test Report.....	32
6.2.14	Hold Stage Review Activity.....	32
7.0	IMPLEMENTATION STAGE.....	33
7.1	Overview	33
7.1.1	Objectives.....	33
7.1.2	Inputs	33
7.1.3	High Level Activities.....	33
7.1.4	Work Products	33
7.2	Tasks and Activities.....	34
7.2.1	Conduct User Training	34

7.2.2	Perform Data Entry or Conversion	34
7.2.3	Establish the System-specific Implementation Environment.....	34
7.2.4	Install/Test the System Software in the Implementation Environment.....	34
7.2.5	Evaluate the System in the Production Environment	34
7.2.6	Conduct Security C&A.....	35
7.2.7	Authorize Transition to Full Operations	35
7.2.8	Conduct Post-Implementation Review	35
7.2.9	Create Documentation and Deliverables.....	35
7.2.9.1	Change Implementation Notice	35
7.2.9.2	Delivered System.....	35
7.2.9.3	Change Control Board Decision Document.....	35
7.2.9.4	Project Termination Plan	35
7.2.10	Hold Stage Review Activity.....	36
8.0	OPERATIONS AND MAINTENANCE STAGE	37
8.1	Overview	37
8.1.1	Objectives.....	37
8.1.2	Inputs	37
8.1.3	High Level Activities.....	37
8.1.4	Work Products	37
8.2	Tasks and Activities.....	37
8.2.1	Perform O&M Planning	37
8.2.1.1	Identify System Change Requests.....	37
8.2.1.2	Perform Release Planning	38
8.2.1.3	Update/Create Planning Documents	38
8.2.2	Perform O&M Requirements Analysis	38
8.2.2.1	Analyze SCR Requirements	38
8.2.2.2	Prepare Functional Requirements Document	38
8.2.3	Perform O&M Design	38
8.2.3.1	Perform System Design.....	38
8.2.3.2	Create/Update Documentation and Deliverables	38
8.2.3.3	Conduct Critical/System Design Review.....	39
8.2.4	Perform O&M Development/Integration & Test.....	39
8.2.4.1	Perform Coding and Unit Testing.....	39
8.2.4.2	Conduct Testing.....	39
8.2.4.3	Integrate and Test Software.....	39
8.2.4.4	Conduct System Test.....	39
8.2.4.5	Conduct Security Test.....	39
8.2.4.6	Conduct Acceptance Test	39
8.2.4.7	Create/Update Documentation and Deliverables	39
8.2.5	Perform O&M Implementation.....	39
8.2.6	Perform Routine Maintenance	40
8.2.6.1	Maintain Data/Software Administration	40
8.2.6.2	Maintain System / Software	41
8.2.7	Create Documentation and Deliverables.....	41
8.2.7.1	In-Process Review Report.....	41
8.2.7.2	User Satisfaction Review Report.....	41
8.2.8	Hold Stage Review Activity.....	41
9.0	DISPOSITION STAGE	42
9.1	Overview	42
9.1.1	Objectives.....	42
9.1.2	Inputs	42
9.1.3	High Level Activities.....	42

9.1.4	Work Products	42
9.2	Tasks and Activities.....	42
9.2.1	Archive or Transfer Data	42
9.2.2	Archive or Transfer Software Components	43
9.2.3	Archive Life cycle Deliverables	43
9.2.4	End the System in an Orderly Manner	43
9.2.5	Dispose of Equipment	43
9.2.6	Conduct Post-Termination Review	43
9.2.7	Create Documentation and Deliverables	43
9.2.7.1	Disposition Plan	43
9.2.7.2	Post-Termination Review Report	43
9.2.7.3	Archived System	43
9.2.8	Hold Stage Review Activity	44
10.0	SDLC TAILORING FOR INDIVIDUAL PROJECTS	45
10.1	Requirements versus Guidelines	46
10.2	Definition of a Project	46
10.3	Classification Schema.....	46
10.4	SDLC Tailoring Approval Process.....	48
10.5	Life cycle Strategies	48
10.5.1	Waterfall	48
10.5.2	Evolutionary	49
10.5.3	Incremental	50
10.5.4	Spiral	51
10.5.5	Agile	52
10.5.6	Advantages/Disadvantages of each Life Cycle Strategy	53
APPENDIX A:	GLOSSARY	55
APPENDIX B:	ACRONYMS	64
APPENDIX C:	TEMPLATES	66
 TABLE OF FIGURES		
Figure 1-1.	SDLC Stages.....	3
Figure 1-2.	Core SDLC Phases and Work Products	6
Table 10-1.	Life Cycle Artifacts by Project Class	48
Figure 10-5.	Waterfall Model	49
Figure 10-6.	Evolutionary Development Model.....	50
Figure 10-7.	Incremental Development Model.....	51
Figure 10-8.	Spiral Development Model.....	52
Figure 10-9.	Agile Development Model	53
Figure 10-10.	Alternative Life cycle Strategies	54

REVISION HISTORY

Version Number	Description	Date
Draft Version 1.1	Draft for Initial Review	March 2005
Draft Version 1.2	Revisions incorporated from Initial Review	May 2006
Draft Version 1.3	Revisions to Incorporate 10 Phases	August 2006
Draft Version 1.4	Revisions to incorporate 11 Stages with corrections for organization titles, and additional figure insertion	March 2006
Draft Version 1.5	Release to support beta training	April 2006
Draft Version 1.6	Revisions to address comments from beta training	May 2006
Draft Version 1.7	Draft with updates to organization and systems nomenclature and reduction to 8 stages	November 2006
Draft Version 1.8	Eliminate contractor reference and correct minor errors	January 2007
Draft Version 1.9	Added GSA PIA reference and template	February 2007
Version 2.0	Rewrote the SDLC. Major changes include: <ul style="list-style-type: none"> ▪ General editing for stylistic consistency and readability ▪ Augmented O&M stage ▪ Simplified presentation of tailoring options ▪ Edited references to Capital Planning, Enterprise Architecture and Security. 	June 2009
Version 2.1	Revisions include: <ul style="list-style-type: none"> • Added FAS PMO organization • Refreshed Security Deliverables • Incorporated classification schema • Added PMP-Lite template for smaller projects • Added Agile Methodology Definition 	February 2012

EXECUTIVE SUMMARY

The Systems Development Life cycle

The Systems Development Life cycle (SDLC) is a framework from which individual IT projects can construct tailored project life cycles appropriate for project characteristics. Though the SDLC is certainly concerned with life cycle activities, it is most concerned with the information developed along a project life cycle, packaged as work products or documents, and the role this information plays in project management, technical, and investment oversight decision-making. The SDLC emphasizes decision processes that influence system cost and usefulness. These decisions must arise from the full consideration of business processes, functional requirements, and economic and technical feasibility in order to produce an effective system.

This SDLC establishes a logical order of events for conducting system development that is controlled, measured, documented, and ultimately improved. As project life cycles are established, project-specific definition and execution of processes is ongoing and the project moves through its defined life cycle. Project participants should evaluate this SDLC as well as their own processes for lessons learned and opportunities for improvement. Each *end-of-stage* review should examine project performance at the conclusion of each stage for such opportunities.

This document does not prescribe a single method applicable without change to every system. Because there is wide variance in the methods, techniques and tools used to support the evolution of systems, and project scopes vary greatly, the SDLC presents guidance for selecting appropriate methods, techniques, and tools based on project characteristics.

One methodology does not fit all sizes and types of system development efforts. Therefore, the Federal Acquisition Service (FAS) SDLC methodology provides for a core set of life cycle stages and products, supported by guidance in how to apply those stages and products for representative project types. Section 10, SDLC Tailoring for Individual Projects, includes descriptions of project types supportive of new system development, operations and maintenance, and micro/internal projects — as well as other tailoring guidance.

Purpose, Scope, and Applicability

The SDLC serves as the mechanism to assure that IT systems support the FAS mission functions and comply with the existing regulation, guidance, and policies. It provides a structured approach to managing information technology (IT) projects beginning with establishing the justification for initiating a systems acquisition, development or maintenance effort and concluding with system disposition. Examples of documentation templates and outlines are included in Appendix C.

The SDLC is applicable to all IT Projects within the FAS OCIO – regardless of whether it is development of a new system or continued maintenance to an existing system.

Changes to this Document

Please direct any changes to this document to the FAS Chief Information Office (CIO), Program Management Office (PMO).

SDLC Objectives

The primary purpose of this guide is to disseminate proven practices to system developers, Project Managers, program/account analysts and system owners/users throughout FAS. The specific objectives include the following:

- To reduce the risk of project failure
- To consider system and data requirements throughout the entire life of the system
- To identify technical and management issues early
- To disclose all life cycle costs to guide business decisions
- To foster realistic expectations of what the systems will and will not provide

- To provide information to better balance programmatic, technical, management, and cost aspects of proposed system development or modification
- To encourage periodic evaluations to identify systems that are no longer effective
- To measure progress and status for effective corrective action
- To support effective resource management and budget planning
- To consider meeting current and future business requirements
- To consider make, buy or enhance alternatives

Key Principles

This guidance document refines traditional information system life cycle management approaches to reflect the principles outlined in the following subsections. These are the foundations for life cycle management.

- **Life cycle Management should be used to ensure a Structured Approach to Information Systems Acquisition, Development, and Operations and Maintenance**

This SDLC describes an overall structured approach to information management. The primary emphasis is on the information and systems decisions to be made and the proper timing of decisions. The SDLC provides a flexible framework for approaching a variety of systems projects. The framework enables system developers, Project Leads, program/account analysts, and system owners/users to combine activities, processes, and products, as appropriate, and to select the tools and methodologies best suited to the unique needs of each project.

- **Support the use of an Integrated Project Team**

The establishment of an Integrated Project Team (IPT) can aid in the success of a project. An IPT is a multidisciplinary group of people who support the Program Manager in the planning, execution, delivery and implementation of life cycle decisions for the project. The IPT is composed of qualified empowered individuals from all appropriate functional disciplines that have a stake in the success of the project. Working together in a proactive, open communication, team oriented environment can aid in building a successful project and providing decision makers with the necessary information to make the right decisions at the right time.

- **Each System Project must have a Program Sponsor**

To help ensure effective planning, management, and commitment to information systems, each project must have a clearly identified Program Sponsor. The Program Sponsor serves in a leadership role, providing guidance to the project team and securing, from senior management, the required reviews and approvals at specific points in the life cycle. The Program Sponsor is responsible for identifying who will be responsible for formally accepting the delivered system at the end of the Implementation Stage. An approval from senior management is required after the completion of the first nine of the SDLC stages, annually during the *Operations and Maintenance Stage* and six-months after the *Disposition Stage*. Approval Authority varies depending on dollar value, visibility level, Congressional interests or a combination of these.

- **Select a Single Project Lead for Each System Project**

The Project Lead has responsibility for the success of the project and works through a project team and other supporting organization structures, such as working groups or user groups, to accomplish the objectives of the project. Regardless of organizational affiliation, the Project Lead is accountable and responsible for ensuring that project activities and decisions consider the needs of all organizations with an interest in the project. The Project Lead develops a project charter to define and clearly identify lines of authority between and within the agency's executive management, program sponsor, user/customer, and developer for purposes of management and oversight.

- **A Comprehensive Project Management Plan is required for Each System Project**

The project management plan is a pivotal element in the successful solution of an information management requirement. The project management plan must describe how accomplishment of each life cycle stage suits the specific characteristics of the project—in other words; it must describe how or if any SDLC tailoring is underway and how it complies with the SDLC. The project management plan is a vehicle for documenting the project scope, tasks, schedule, allocated resources, and interrelationships with other projects. The plan provides direction for the many activities of the life cycle and may undergo periodic expansion or refinement during each stage of the life cycle.

- **Assign Specific Individuals to Perform Key Roles throughout the Life cycle**

Certain roles are vital to a successful system project and at least one individual should be responsible for each key role. Assignments may be on a full- or part-time basis as appropriate. Key roles include program/functional management, quality assurance, security, telecommunications management, data administration, database administration, logistics, financial, systems engineering, test and evaluation, contracts management, and configuration management. For most projects, the Program Sponsor will designate more than one individual to represent the actual or potential users of the system.

- **Obtaining the Participation of Skilled Individuals is Vital to the Success of the System Project**

The skills of the individuals participating in a system project are the single most significant factor for ensuring the success of the project. The intent SDLC manual is not to replace information management skills or experience. There is considerable discussion in many of the skills required for a system project in later sections. This discussion does not obviate the fact that the required skill combination will vary according to the project. All individuals participating in a system development project are encouraged to obtain assistance from experienced information management professionals.

- **Documentation of Activity Results and Decisions for Each Stage of the Life cycle are Essential**

Effective communication and coordination of activities throughout the life cycle depend on the complete and accurate documentation of decisions and the events leading up to them. Undocumented or poorly documented events and decisions can cause significant confusion or wasted efforts and can intensify the effect of turnover of project management staff. No activity will be complete, nor decisions made, until there is tangible documentation of the activity or decision. For some large projects, advancement to the next stage cannot occur until the senior management reviews and approves stage work products.

- **Data Management is required Throughout the Life cycle**

The FAS considers the data processed by systems to be an extremely valuable resource. Accurate data is critical to support organizational missions. The volume of data FAS systems manage and the trend toward interfacing and sharing data across systems and programs, underscores the importance of data quality. Systems life cycle activities stress the need for clear definition of data, the design and the implementation of automated and manual processes that ensure effective data management.

- **Each System Project Must Undergo Formal Acceptance**

The Program Sponsor identifies the representative who will be responsible for formally accepting the delivered system at the end of the Implementation Stage. The Program Sponsor, or his designee, formally accepts the system by signing an Implementation Stage Review and Approval Certification along with the developer.

- **Consultation with Oversight Organizations Aids in the Success of a System Project**

A number of FAS oversight bodies, as well as external organizations, have responsibility for ensuring that information systems activities comply with GSA/FAS/Federal guidance and standards and available resources used effectively. Each project team should work with these organizations, as appropriate, and encourage their participation and support as early as possible in the life cycle to identify and resolve potential issues or sensitivities and thereby avoid major disruptions to the project. Project personnel should assume that all documentation is subject to review by oversight bodies.

- **A System Project may not Proceed Until Resource Availability is Assured**

Beginning with the approval of the project, the continuation of a system is contingent on a clear commitment from the Program Sponsor. This commitment is embodied in the assurance that the necessary resources will be available, not only for the next activity, but as required for the remainder of the life cycle.

- **Incorporate Security into all phases of the Life cycle**

Including security early in the SDLC and throughout the life cycle will result in less expensive and more effective security than adding it at the end to an operational system. Each phase of the FAS SDLC will include a minimum set of security steps needed to effectively incorporate security into a new system.

- **Each System Project Must Comply with the GSA Enterprise Architecture (EA).**

The GSA EA provides the framework through which business and IT are aligned to enable business agility through planning and implementing of flexible IT solutions that are easily adaptable to continuously evolving business strategies and organizational goals. The GSA EA is in full support of the Federal Enterprise Architecture which can be found at:

<http://www.whitehouse.gov/omb/e-gov/fea/>

1.0 INTRODUCTION

1.1 Background

The Federal Acquisition Service (FAS) spends millions of dollars each year on the acquisition, design, development, implementation, and maintenance of information systems vital to mission programs and administrative functions. The need for safe, secure, and reliable system solutions is a result of the increasing dependence on computer systems and technology to provide services and develop products, administer daily activities, and perform short- and long-term management functions. There is also a need to ensure privacy and security when developing information systems, to establish uniform privacy and protection practices, and to develop acceptable implementation strategies for these practices.

The FAS needs a systematic and uniform methodology for information systems development. Using this SDLC will ensure that systems developed by the FAS meet IT mission objectives; are compliant with the current and planned Enterprise Architecture (EA); and are easy to maintain and cost-effective to enhance. Sound life cycle management practices include planning and evaluation in each stage of the information system life cycle. The appropriate level of planning and evaluation is commensurate with the cost of the system, the stability and maturity of the technology under consideration, how well defined the user requirements are, the level of stability of program and user requirements and security considerations.

1.2 Purpose

This SDLC methodology establishes procedures, practices, and guidelines governing the concept approval, project planning, requirements analysis, design, development, integration and test, implementation, and operations, maintenance and disposition of information systems within the FAS. It fully concurs with existing GSA and FAS policy and guidelines for strategic planning, budgeting, Capital Planning and Investment Control (CPIC), Enterprise Architecture (EA), acquisition, procurement, and deployment.

1.3 Scope

The SDLC applies to all FAS information systems and applications – regardless of whether it is a new systems development effort, on-going maintenance of an existing system, or development of a small system internal to the FAS OCIO. The specific participants in the life cycle process, and the necessary reviews and approvals, vary from project to project. Section 10 of this document provides guidance for the tailoring the SDLC to individual projects, based on cost, complexity, and criticality to FAS' mission.

Appendix C contains templates for each of the documents identified as SDLC products, with guidance for both format and content. With prior approval, the packaging and the level of detail is adaptable to the needs of the individual project. For example, some documents may be combined (e.g., a single planning document instead of many) or the form of individual documents may be varied (e.g., emails or memoranda instead of formal documents).

1.4 Assumptions

The following lists the assumptions for the SDLC and the scope/purpose of this document:

- The Project Manager is well versed in Project Management best practices (e.g., Project Management Institute (PMI) Project Management Body of Knowledge (PMBOK). The SDLC does not specifically address these areas and practices
- The SDLC describes what must be done and does not get into the “how”. It is assumed that the project team is well versed in the software engineering discipline. If the reader requires additional information in the how, they are to explore other options for gaining this knowledge.
- The SDLC describes what must be done as it relates to software development. There are many OMB/GSA/FAS requirements for projects (e.g., Capital Planning and Investment Control (CPIC), Enterprise Architecture, Earned Value Management (EVM), IT Governance) and the SDLC does not get into the specifics of each of these areas.
- The SDLC applies to all projects within the FAS OCIO including all Development, Maintenance and Enhancement (DME) projects; and Operations and Maintenance (O&M) projects.

- The SDLC embraces minimizing the documentation requirements for various projects. This can be accomplished by numerous means including:
 - Creating general program plans that apply to multiple projects
 - Reusing documents across projects
 - Combining multiple documents into one as described by the tailoring guidelines in Section 11
 - Creating templates and only updating/issuing the information that is pertinent to the specific project
- The SDLC recognizes that the majority of projects within the FAS OCIO are Operations and Maintenance (O&M) with enhancements/updates being released on a regular basis. The O&M phase specifically addresses this scenario.
- Because of the wide variety of projects/programs within the FAS OCIO and the interrelationship between government personnel and contractors supporting these projects/programs, the SDLC does not address the following items. Instead, it is assumed that each of these will be addressed by the various project plans:
 - Roles/responsibilities of the project team members to include contractors
 - Roles/responsibilities of organizations external to the FAS OCIO (e.g., Business lines, user groups, other GSA organizations, etc.)
 - Who will be performing specific tasks
 - Governance structures to include Configuration Control Boards (CCB), Integrated Project Team (IPT), Program Management Office (PMO), various review boards
 - Organization charts

1.5 SDLC Overview

The SDLC provides the framework for defining the activities that should be completed as part of the project. Projects receive approval and evolve in tailored compliance to the SDLC and business cases, successively validated, proceed toward assigned objectives.

1.5.1 SDLC Stages

The SDLC is comprised of eight stages, during which defined IT work products move towards completion. Each stage is divided into activities and tasks, and has a measurable end point. The execution of all eight stages is based on the premise that the quality and success of the project depends on a feasible concept, comprehensive and participatory project planning, commitments to resources and schedules, complete and accurate requirements, a sound design, consistent and maintainable construction techniques, and a comprehensive testing program.

Descriptions of the activities and work products for each SDLC stage are described in subsequent Sections. Typically, projects will move through every stage, generally from left to right; however, some stages and/or sequence of stages occur more than once before full system capability deployment. Nonetheless, each time a stage executes, the activities, products, and stage review requirements must complete in an appropriately tailored manner. Section 10, Tailoring, addresses the concept of project life cycles and the definition of life cycle segments which comprise each project life cycle. **Figure 1-1** provides a start-to-end capsule of each stage of the SDLC.

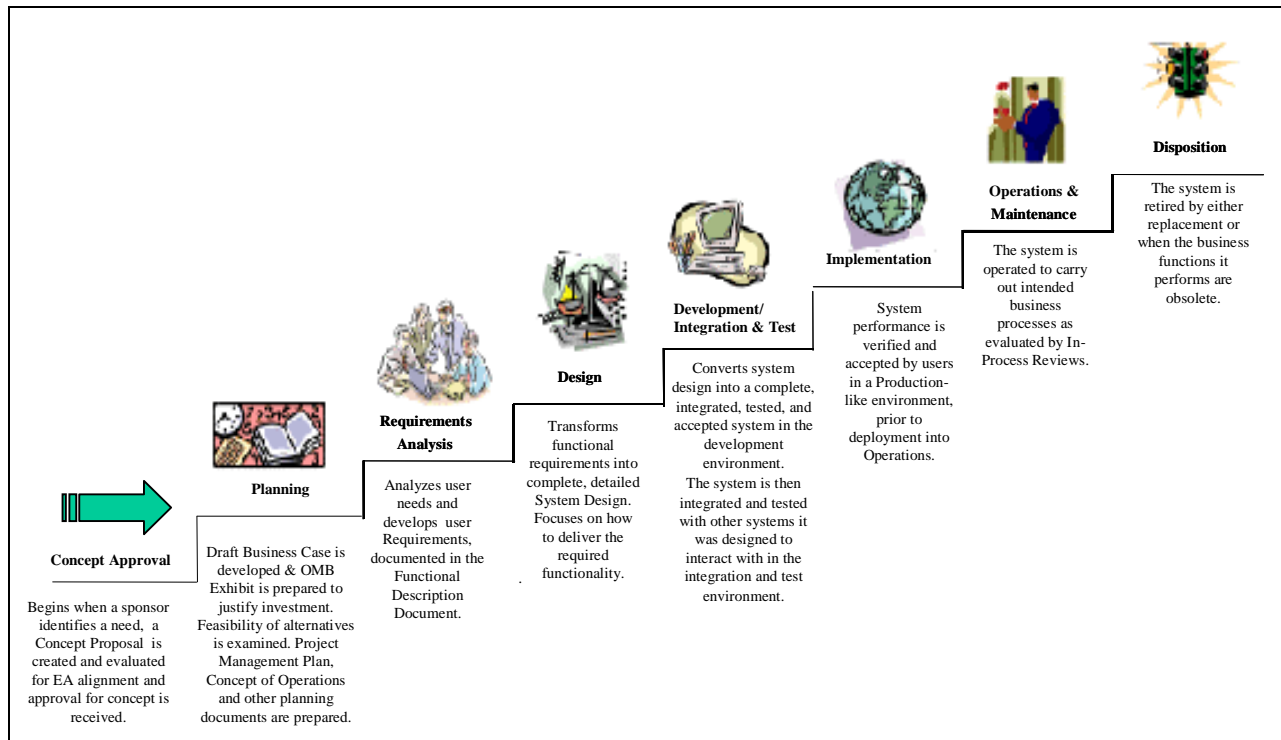


Figure 1-1. SDLC Stages

At the conclusion of each stage, a Stage Review Activity is held to review the work products of that stage and to determine whether to proceed to the next stage, continue work in the current stage, or abandon the project. The approval of the system owner and other project stakeholders at the conclusion of each stage enables both the system owner and project manager to remain in control of the project throughout its life, and prevents the project from proceeding beyond authorized milestones.

The life cycle model provides a method for performing the individual activities and tasks within an overall project framework. The stages and activities are designed to follow each other in an integrated fashion, whether the stages of development are accomplished sequentially, concurrently, or cyclically.

The following subsections provide a brief overview of each SDLC stage. Each stage is discussed in more detail in Section 2 – Section 9.

1.5.1.1 Concept Approval Stage

Recognition of an investment opportunity occurs with the identification and validation of a business need. The Program may designate a Program Manager to manage the opportunity and document the business need as a Draft Business Case (DBC) that, in turn, forms the basis for a determination whether there is sufficient justification to pursue developing the concept. During this stage, Security Risk Assessment and Enterprise Architecture business alignment and technical compliance reviews begin, in order to ensure that the proposed need or opportunity is appropriate and aligned with the GSA Enterprise Architecture. Regular review of EA alignment occurs throughout the SDLC to ensure the system remains compliant with the target EA.

1.5.1.2 Project Planning Stage

The approved Draft Business Case (DBC) is updated to identify the scope of the system and include all the elements of a business case necessary to support the OMB Exhibit 300 preparation. The DBC captures the business functions, goals and objectives that the IT project will satisfy. It also captures critical success factors, assumptions and constraints as well as performance measures. As appropriate, alternative solutions undergo analysis for feasibility and appropriateness. An initial plan for tailoring the SDLC to project needs and acquisition strategy are significant components of project planning.

A project may warrant a separate OMB Exhibit 300/business case depending on dollar value, visibility level, congressional interests or a combination of these. An OMB Exhibit 300 presents the business case in a level of detail appropriate to the associated level of review and approval, as determined based on project characteristics.

During the planning stage, the system concept is further developed into a Concept of Operations Definition (ConOps). The ConOps defines the scope and characteristics of the proposed system (from the user's perspective) and the operational environment in which it needs to function. The high level requirements are analyzed, system concepts synthesized, concepts evaluated (in terms of cost, mission and environmental impacts), and the best system concept(s) selected and described. The optimum capabilities resulting from the trade-off analyses are documented in the ConOps describing how the business will operate once the proposed system is implemented, and to assess how the system will impact employee and customer privacy.

The preliminary planning of the IT Project performed during the Project Planning Stage while developing the Draft Business Case is finalized in the Project Management Plan (PMP). Other supporting planning documents (e.g., Quality Assurance Plan, Risk Management, and Configuration Management Plan) are prepared during this stage. The PMP will also define the management controls in place to ensure the readiness of products and/or services to provide the required capability on time and within budget, project resources, project life cycle, activities, products, schedules, tools, and reviews are defined.

The Project Management Plan must describe the project approach to tailoring the SDLC for stage activities, products, schedules, and reviews. The planning documents describe the distribution of activities/products of the SDLC among government and contractor organizations and provide for sufficient programmatic integration to ensure that the multiple contributions will be successfully integrated into the final solution.

A Stage Review Activity is held to constitute Program Authorization and approval to move to the next stage. The Stage Review Activity concludes the Project Planning Stage, establishes the Project Baseline and authorizes movement into the Requirements Analysis stage.

1.5.1.3 Requirements Analysis Stage

The Draft Business Case and Concept of Operations are the basis for developing the Functional Requirements Document (FRD). The FRD formally describes system requirements in terms of system function, data, performance, security, and maintainability requirements. Interface requirements are defined with respect to external systems and less formally between major subsystems within the system. Requirements are defined to a level of detail sufficient for systems design to proceed. Each requirement must be measurable, testable, and traceable to a source requirement in the products developed during prior stages. Any privacy issues are addressed to ensure issuance of Privacy Act Notices as required and execution of a Privacy Impact Assessment as appropriate. A Systems Requirements Review approves work products, establishes the Functional Baseline, and authorizes the project to proceed to the Design Stage.

1.5.1.4 Design Stage

The architecture and physical characteristics of the system are designed during this stage. The operating environment is established, major subsystems and their inputs and outputs are defined, and processes are allocated to resources. Everything requiring user input or approval must be documented and reviewed by the user. The physical characteristics of the system are specified and a detailed design is prepared. Subsystems identified during design are used to create a detailed structure of the system. Each subsystem is partitioned into one or more design units or modules. Detailed logic specifications are prepared for each software module. As actual system design is taking place, preparations for Implementation and Operations stages are initiated and a formal Security Risk Assessment is conducted. This stage concludes with a Critical Design Review, which approves work products, establishes the Allocated Baseline and authorizes the project to proceed to the Development Stage.

1.5.1.5 Development/Integration and Test Stage

The detailed specifications produced during the design stage are translated into hardware, communications, and executable software. Software modules are unit tested and successively

integrated/tested in a systematic manner, until entire subsystems have been tested. Hardware is acquired/assembled and tested. Integration plans are prepared to guide integration activities in the next stage. A Test Readiness Review is conducted to assess work products and authorize the project to proceed into Integration and Test.

During Integration and Test the various software and hardware subsystems/components of the system are integrated and systematically tested. Integration, performance, security, and functional testing occur in preparation for users to perform acceptance testing of the system. The user tests the system to ensure that the functional requirements, as defined in the functional requirements document, are satisfied by the developed or modified system. Prior to installing and operating the system in a production environment, the system must undergo IT Systems Security Certification and Accreditation activities. An Implementation Readiness Review is conducted to assess work products, establish the Product Baseline, and authorize the project to proceed into Implementation.

1.5.1.6 Implementation Stage

The system or system modifications are installed and made operational in a production environment. The stage is initiated after the system has been tested and accepted by the user in the integration and test environment. This stage continues until the system is operating in the production (or near-production) environment in accordance with the defined user requirements. A Post-Implementation Review (PIR) is conducted to validate that the system satisfies user requirements. A Production Readiness Review is conducted to assess completion of Implementation Stage activities and marks the system as ready for transition into the Operations and Maintenance stage. The completion of the Implementation Stage marks the end of the IT project(s) spawned after program authorization, with the Post-Implementation Review validating that the solution satisfies original user requirements and updating the Production Baseline being maintained by the CCB.

1.5.1.7 Operations and Maintenance Stage

The Operations and Maintenance Stage is ongoing. The system is monitored for continued performance in accordance with user requirements, and needed system modifications are incorporated. The operational system is periodically assessed through In-Process Reviews to determine how the system can be made more efficient and effective. Operations continue as long as the system can be effectively adapted to respond to an organization's needs. When significant and/or unfunded modifications are identified as necessary, the modifications reenter the SDLC at the O&M Planning. The Operations and Maintenance Stage continues until a decision to retire the system is reached, at which time the system enters the Disposition Stage.

1.5.1.8 Disposition Stage

The Disposition Stage ensures the orderly termination of the system, preserving the vital information about the system so that some or all of the information may be reactivated in the future if necessary. Particular emphasis is given to proper preservation of the data processed by the system, so that the data is effectively migrated to another system or archived in accordance with applicable records management regulations and policies, for potential future access.

1.6 Documentation Requirements

This life cycle framework specifies which documents should be generated during each stage. The amount the project and system documentation required throughout the life cycle depend on the size and scope of the project. The requirements for documentation should not be interpreted as mandating formal, standalone, printed documents in all cases.

Once developed, some work products remain unchanged thereafter while others are subsequently revised, evolving through later SDLC stages. The core set of SDLC Stages and associated work products are identified in **Figure 1-2**. Templates for documents that originate in this SDLC are found in Appendix C. Sources for templates appropriate to some of the SDLC products, including some relating to Acquisition, CPIC, Security, Privacy, etc., are referenced, as appropriate, from the text.

Core Phases and Products	FAS SDLC Stages	Concept Approval	Planning	Requirements Analysis	Design	Development/ Integration & Test	Implementation	Operations & Maintenance	Disposition
Life Cycle Work Products	Product Source								
Draft Business Case (DBC)	SDLC C-1	C	F	*	*	*	*	*	
Security Risk Assessment	SAISO Office	C	R	F	*	*	*		
Risk Management Plan (RMP)	SDLC C-4	c	C	R	R	F		*	*
Cost-Benefit Analysis (CBA)	SDLC C-2		C	R	R	F			
Feasibility Study	SDLC C-3		C	F					
Business Case (OMB Exhibit)	GSA ITPM Office		C/F	*	*	*	*	*	*
Concept of Operations (ConOps)	SDLC C-7		C	R	R	R	F	*	*
Project Management Plan (PMP)/PMP-Lite	SDLC C-24/C-25	c	C	R	R	F	*		
Configuration Management Plan (CMP)	SDLC C-5		C	R	R	F	*	*	
Quality Assurance Plan (QAP)	SDLC C-6		C	R	R	F	*	*	
System Security Plan (SSP)	SAISO Office		C	R	R	F	*	*	
Acquisition Plan	GSA APW		C	R	R	F	*	*	
System Engineering Management Plan (SEMP)	SDLC C-8		C/F	*	*	*	*	*	*
Functional Requirements Document (FRD)	SDLC C-9			C	F				
Requirements Traceability Matrix (RTM)	SDLC C-28			C	R	F			
Interface Control Document (ICD)	SDLC C-11			C	R	F	*	*	
Privacy Act Notice/Privacy Impact Assessment	SAISO Office		C	C	F		V		
Test Plan (PT)	SDLC C-10		C	C	R	F	*	*	
Conversion Plan	SDLC C-12				C	F	*		
System Design Document (SDD)	SDLC C-13				C	F		*	
Implementation Plan (IMP)	SDLC C-14				C	F			
Maintenance Manual (MM)	SDLC C-15				C	F	*	*	
Operations Manual (OM) (System Administration Manual)	SDLC C-16/17				C	F	*	*	
Training Plan (TP)	SDLC C-18				C	F	*	*	
User Manual (UM)	SDLC C-19				C	F	*	*	
IT Contingency Plan	SDLC C-20				C	F	*	*	
Software Development Document/Folder (SDF)	Project					C/F	*	*	
System Software	Project					C/F	*	*	
Test Files/Data	Project					C/F		*	
Test Analysis Report (TAR)	SDLC C-21					P	*	*	
Test Problem Report	Project					P	P	P	
User Acceptance Test Report	SDLC C-26					P			
*IT Systems Security Certification & Accreditation	SAISO Office					C/F			
Security Vulnerability Scan	ISSO/SecWiki					C/E	F	*	
Delivered System	Project						C/F	*	
Change Implementation Notice (CIN)	Project						C/F	*	
Post-Implementation Review (PIR)	SDLC C-22						C/F	*	
In-Process Review Report (IPR)	Project							P	
User Satisfaction Report	Project							P	
Disposition Plan	SDLC C-23								C/F
Post-termination Review Report	Project								P
Archived System	Project								C/F

SDLC C-n Product Template in SDLC Appendix
 External Product Template defined external to SDLC
 Project Product Template defined by the project

KEY: C=Create, E=Execute, F=Finalize, M=Monitor, P=Produce, R=Revise, V=Validate, *=Update if needed, lower case=optional

Figure 1-2. Core SDLC Phases and Work Products

1.7 Key Activities Related to System Development

This section describes the key activities that are part of any systems development effort but are not described in detail in the SDLC because these topics, in general, are expansive and there are numerous references which describe them in more detail.

1.7.1 Strategic Planning

Strategic planning provides a framework for analyzing where an enterprise is and where an enterprise should be in the future. In the Federal government agency strategic plans are required by the Government Performance and Results Act (GPRA). Agency strategic plans provide the framework for implementing all other parts of this Act, and are the key part of the effort to improve performance of government programs and operations.

The GSA Strategic Plan guides the annual budget and performance planning in all the offices and SSOs. It sets the framework for measuring progress and ensuring accountability to the public. The strategic plan identifies goals, objectives and strategies in support of the Agency's mission and vision. This, in turn, drives the IT strategic plan across GSA, ensuring linkage to the overall goals and direction the Administrator has set for the Agency. Strategic planning is not part of the SDLC, but plays a significant role in selecting IT projects to be initiated and continued.

1.7.2 Capital Planning and Investment Control (CPIC)

The CPIC process implements the FAS' information technology capital planning and investment control process. The CPIC process uses the "Select-Control-Evaluate" methodology recommended by OMB, GAO guidance to implement the strategic and performance directives of the Clinger-Cohen Act, GSA/FAS Management Directives, and other statutory provisions affecting information technology investments. The process complements the SDLC process by providing fiscal oversight of system development projects and linking IT investment decisions to GSA/FAS strategic goals and objectives.

1.7.3 Systems Security

The Federal Government has become increasingly reliant on IT systems to support day-to-day and critical operations/business transactions. Risks to system and data confidentiality, integrity, and availability can impact an organization's ability to execute its mission and business strategy. To minimize the impact associated with these risks, federal IT security policy requires all IT systems to be certified and accredited prior to being placed into operation in accordance with the requirements of OMB A-130. The Federal Information Security Systems Management Act of 2002 (FISMA) requires agencies to have plans for information security programs to assure adequate information security for networks, facilities, information systems, or groups of information systems, as appropriate.

The GSA IT Security Procedural Guide: Managing Enterprise Risk (Security Categorization, Risk Assessment, & Certification and Accreditation - CIO-IT Security-06-30) Handbook¹ provides managers with a single source of information for conducting certification and accreditation and securing IT Systems. Additional guidance can be found in the series of CIO-IT Security Procedural Guides, and NIST Special Publications. These guides offer guidance for implementing security processes and controls and templates to meet reporting requirements. Templates are provided for the Security Risk Assessment, Systems Security Plan, Contingency Plan, and Certification and Accreditation (C&A) memoranda. The C&A process is an integral part of the SDLC and the CPIC process.

1.7.4 Enterprise Architecture

The development of information technology architectures is a requirement of the Clinger-Cohen Act. GSA is building an enterprise architecture that promotes the effective management and operation of IT investments and services.

¹ GSA Order CIO HB 2100.1C, GSA Information Technology Security Policy

² NIST Special Publications

An enterprise architecture (EA) provides a comprehensive, integrated picture of current capabilities and relationships (i.e., the current architecture), an agreed upon blueprint for the future (i.e., the target architecture), and a strategy for transitioning from the current to the target environment. EA products describe the information needed to carry out key business functions and processes; identify the system applications that create or manipulate data to meet business information needs; and documents the underlying technologies (i.e., hardware, software, communications networks, and devices) that enable the generation and flow of information.

The EA is an essential tool for strategic planning, managing IT resources, and making maximum use of limited IT dollars. The EA also ensures the alignment of IT with GSA's strategic goals so that business needs drive technology rather than the reverse. The EA also assists to:

- Identify redundancies, and thus potential cost savings;
- Highlight opportunities for streamlining business processes and information flows;
- Optimize the interdependencies and interrelationships among the programs and services of the Agency's various component organizations as well as with external agencies;
- Ensure a logical and integrated approach to adopting new technologies;
- Promote adherence to Agency-wide standards including those for systems security; and

1.7.5 Performance Management

Performance measurement is an essential element in developing effective systems through a strategic management process. FAS' mission, goals, and objectives are identified in its Strategic Plan. Strategies are developed to identify how FAS can achieve these goals. For each goal, FAS establishes a set of performance measures. These measures enable FAS leadership to assess how effective each of its projects are in improving Service operations.

For FAS to make this assessment, the current performance level for each measure (performance level baseline) for the existing systems must be determined. For each project plan, as part of the cost benefit analysis, estimates for the performance levels expected to be attained as a result of the planned improvements are made. As the project's improvements are implemented, actual results are compared with the estimated gains to determine the success of the effort. Further analysis of the results may suggest additional improvement opportunities.

Performance Measurement and evaluation is the principal method for determining realization of identified benefits.

1.7.6 Business Process Re-engineering

The primary underpinning of any new system development or initiative should be business process reengineering. Business process reengineering (BPR) involves a change in the way an organization conducts its business. BPR is the redesign of the organization, culture, and business processes using technology as an enabler to achieve quantum improvements in cost, time, service, and quality. Information technology is not the driver of BPR. Rather, it is the organization's desire to improve its processes and how the use of technology can enable some of the improvements.

BPR may not necessarily involve the use of technology. When BPR is applied to one or more related business processes, an organization can improve its products and services and reduce resource requirements. The results of a successful BPR program are increased productivity and quality improvements. BPR is not just about continuous, incremental and evolutionary productivity-enhancements. It also utilizes an approach that suggests scraping a dysfunctional process and starting from scratch to obtain larger benefits.

2.0 CONCEPT APPROVAL STAGE

2.1 Overview

2.1.1 Objectives

The Concept Approval Stage begins when management identifies a business need that can be satisfied by the application of information technology. The objectives of the Concept Approval Stage are to:

- Identify and validate an opportunity to improve business accomplishments of the organization or a deficiency related to a business need.
- Identify significant assumptions and constraints on solutions to that need.
- Recommend the exploration of alternative concepts and methods to satisfy the need.

IT projects may be initiated as a result of business process improvement activities, changes in business functions, advances in information technology, or may arise from external sources, such as public law, the general public or state/local agencies. The Program Sponsor articulates this need within the organization to initiate the project life cycle. During this stage, a Program Manager is appointed who prepares a Draft Business Case (DBC). When an opportunity to improve business/mission accomplishments or to address a deficiency is identified, the Program Manager documents these opportunities in the DBC.

2.1.2 Inputs

- Business Opportunity
- Enterprise Architecture
- System Concept

2.1.3 High Level Activities

- I). Identify the Opportunity to Improve Business Functions
- II). Identify a Program Sponsor
- III). Create Documentation and Deliverables
- IV). Hold Stage Review Activity

2.1.4 Work Products

- Draft Business Case (DBC)
- Enterprise Architecture Alignment Summary

2.2 Tasks and Activities

The following activities are performed as part of the Concept Approval Stage. The results of these activities are captured in the Draft Business Case. For every approved IT project, the agency should designate a responsible organization and assign the organization sufficient resources to execute the project.

2.2.1 Identify the Opportunity to Improve Business Functions

Identify why a business process is necessary and what business benefits can be expected by implementing this improvement. A business case must be established in which a business problem is clearly expressed in purely business terms. Provide background information at a level of detail sufficient to familiarize senior managers to the history, issues and customer service opportunities that can be realized through improvements to business processes with the potential support of IT. This background information must not offer or predetermine any specific automated solution, tool, or product.

2.2.2 Identify a Program Sponsor

The Program Sponsor is the principle authority on matters regarding the expression of business needs, the interpretation of functional requirements language, and the mediation of issues regarding the priority,

scope and domain of business requirement. The Program Sponsor may identify and enlist the assistance of the Program Manager in the execution of this stage as well as subject matter experts from Enterprise Architecture and IT Security.

2.2.3 Create Documentation and Deliverables

2.2.3.1 Prepare Draft Business Case

This describes the need or opportunity to improve business functions. It identifies where strategic goals are not being met or mission performance needs to be improved. Document the business need as the Draft Business Case (DBC) that forms the basis for a determination whether there is sufficient business justification to develop the concept. [Appendix C-1](#) provides a template for the Draft Business Case along with tailoring guidelines for smaller projects. During this stage the PM will only complete the first two sections of the DBC.

2.2.3.2 Analyze Fit with the Enterprise Architecture

A formal assessment of EA alignment is conducted at the beginning of Concept Approval and at selected points through the FAS SDLC process to ensure identification of any variances in planned alignment. The sponsoring organization must establish alignment with GSA architectural principles during the Concept Approval stage and document it in the Enterprise Architecture Alignment Summary which is contained within the DBC.

The Enterprise Architecture Alignment Summary provides the results of EA Alignment and Assessment Reviews that occur periodically throughout the system's life. All technology systems and projects must be aligned with the GSA Enterprise Architecture (EA). Current IT capabilities, planned IT programs, and ongoing projects, need to be understood in terms of how they relate to the GSA Business Architecture, Target Architecture, and high-level transition strategy.

The EA assessment process is integrated throughout the life cycle to ensure that IT programs and projects are initiated, planned, and executed consistent with the GSA target architecture. A formal assessment of EA alignment is conducted beginning at Program Initiation and throughout the SDLC.

2.2.4 Hold Stage Review Activity

At the end of this stage, the Draft Business Case is approved before proceeding to the next stage. The Draft Business Case should convey that this project is a good investment and is aligned with the GSA EA.

Authorization to Proceed includes approvals by the Program Sponsor / Program Manager, CIO (if preliminary EA business compliance is assessed), and the Review/Approval Authority. Approvals should be annotated on the Draft Business Case.

3.0 PLANNING STAGE

3.1 Overview

3.1.1 Objectives

The main objectives of the planning stage are to select the strategies, policies, processes, and procedures for achieving the objectives and goals of the project. Planning is deciding, in advance, what to do, how to do it, when to do it, where to do it, and who is going to do it. Project planning applies to all projects – regardless of their size.

The requirements identified in project related materials (e.g., draft business case) are the primary input to this phase. The level of detail will vary depending on project size. The preparation of documents related to this phase involves several critical planning issues such as the identification of preliminary requirements; staff, schedule, and cost estimates; the technical and managerial approaches that will be used; and the assessment of potential risks associated with the project. This information forms the foundation for all subsequent planning activities.

Many of the plans essential to the success of the entire investment program and individual IT projects are created in this stage; the created plans are then reviewed and updated throughout the remaining SDLC stages. In the Planning Stage, the concept is further developed to describe how the business will operate once the approved system is implemented and to assess how the system will impact employee and customer privacy. To ensure the products and/or services provide the required capability on time and within budget, project resources, activities, schedules, tools, and reviews are defined. Additionally, security certification and accreditation activities begin with identification of system security requirements and the completion of a high-level vulnerability assessment. The Planning Stage begins when the Draft Business Case has been formally approved and resources have been committed by the Program Sponsor to enter the Planning Stage.

The review and approval of the Draft Business Case begins the formal studies and analysis of the need in the Planning Stage and begins the more detailed business analysis and planning of the project.

3.1.2 Inputs

- Draft Business Case
- Enterprise Architecture Alignment Summary

3.1.3 High Level Activities

- I). Form a project organization
- II). Study and analyze the business need
- III). Form the project acquisition strategy
- IV). Create internal processes
- V). Determine security classification
- VI). Analyze project schedule and cost
- VII). Establish agreements with stakeholders
- VIII). Plan the solicitation, selection, and award
- IX). Create Documentation and Deliverables
- X). Hold Stage Review Activity

3.1.4 Work Products

- Draft Business Case (DBC)
 - Cost-Benefit Analysis
 - Feasibility Study

- Security Risk Assessment Report
- System Security Plan
- Concept of Operations
- Acquisition Plan
- Project Management Plan/PMP-Lite
 - Systems Engineering Management Plan
 - Configuration Management Plan
 - Quality Assurance Plan
 - Risk Management Plan
- FAS Earned Value Management Package
- Exception Request Form

3.2 Tasks and Activities

The following activities are performed as part of the Planning Stage. The results of these activities are captured in work products that are initiated during this stage as well as updates to prior stage products.

3.2.1 Form (or appoint) a Project Organization

This activity involves the appointment of a Program Manager (if not previously identified) who carries both the responsibility and accountability for project execution. The Program Manager develops a Program Charter that establishes the scope and responsibilities for the project. For small efforts, the project organization may only involve assigning a project to a manager within an existing organization that already has an inherent support structure. For new, major projects, a completely new organizational element may be formed - requiring the hiring and reassignment of many technical and business specialists.

Each project must have an individual designated to lead the effort. The individual selected will have appropriate skills, experience, credibility, and availability to lead the project. Clearly defined authority and responsibility must be provided to the Program Manager.

The Program Manager will work with stakeholders to identify the scope of the proposed program, participation of the key organizations, and potential individuals who can participate in the formal reviews of the project. This decision addresses both programmatic and information management-oriented participation as well as technical interests in the project that may be known at this time.

In view of the nature and scope of the proposed program, the key individuals and oversight committee members who will become the approval authorities for the project will be identified.

3.2.2 Study and Analyze the Business Need

The project team, identified in the Program Charter, and supplemented by enterprise architecture or other technical experts, as required, should analyze all feasible technical, business process, and commercial alternatives to meeting the business need. These alternatives should then be analyzed from a life cycle cost perspective. The results of these studies should show a range of feasible alternatives based on life cycle cost, technical capability, and scheduled availability. Typically, these studies should narrow the system technical approaches to only a few potential, desirable solutions that should proceed into the subsequent life cycle stages.

3.2.3 Form the Project Acquisition Strategy

The acquisition strategy should be included in the Draft Business Case. The project team should determine the strategies to be used during the remainder of the project concurrently with the development of the Cost Benefit Analysis (CBA) and Feasibility Study. For smaller projects, the CBA and Feasibility Study will be included as part of the DBC. The Acquisition Strategy answers the questions:

- Is there sufficient staff to accomplish the work of the project or will it require additional support from an IT contractor?
- Is there already a contractor in place that will be performing this work and does it require a change to the existing SOW or contract?
- What technology options are available?
- Are there opportunities for reuse of existing technology?
- Are there Commercial Off-the-Shelf solutions available?
- What type of contract will best suit the business and technical requirements of the project?
- What life cycle segments will comprise the project life cycle and what organizations (contractor/ government) will perform those segments?

Refinement of the role of system development contractors will occur during the subsequent stages. For example, one strategy option would include active participation of system contractors in the Requirements Analysis Stage and another contractor for the systems development activities. In this case, the Planning Stage must include complete planning, solicitation preparation, and source selection of the participating contractors (awarding the actual contract may be the first activity of the next stage). If contractors will be used to complete the required documents, up-front acquisition planning is essential.

3.2.4 Create Internal Processes

Create, gather, adapt, and/or adopt the internal management, engineering, business management, and contract management internal processes that will be used by the project office for all subsequent life-cycle stages. This could result in the establishment of teams or working groups for specific tasks, (e.g., quality assurance, configuration management, and testing). Plan, articulate, and gain approval for the resulting processes. These processes will be a tailored version of this SDLC, applied to the specific needs of the project. The products required for this project are determined and appropriate activities are included in project and engineering work break down structures.

3.2.5 Determine Security Categorization

Security categorization, in accordance with the Federal Information Processing Standards Publication 199 (FIPS Pub 199), *Standards for Security Categorization of Federal Information and Information Systems*, should be conducted during this stage. Security categories are based on the potential impact on an organization should certain events occur which jeopardize the information and information systems needed by the organization to accomplish its assigned mission, protect its assets, fulfill its legal responsibilities, maintain its day-to-day functions, and protect individuals. The security category of an information system must also consider the security categories of all information types resident on the information system.

3.2.6 Analyze Project Schedule and Cost

Analyze and refine the project schedule and cost, taking into account risks and resource availability. Develop a detailed schedule for the initial life cycle segments, especially for its Requirements Analysis Stage, as well as for all subsequent segments and stages. Develop a Life cycle Cost Estimate using the information in the Draft Business Case, WBS, and Schedule, plus information from the prior phase, estimate the life cycle cost and document related assumptions and risks.

3.2.7 Establish Agreements with Stakeholders

Establish relationships and agreements with internal and external organizations that will be involved with the project. These organizations may include agency and FAS oversight offices, agency personnel offices, agency finance offices, internal and external audit organizations, and agency resource providers (people, space, office equipment, communications, etc) as well as organizations directly involved with the project (e.g., user communities, hosting, training, approval and sign-off decisions, etc.)

3.2.8 Plan the Solicitation, Selection and Award

During this stage or subsequent stages, as required by the Federal Acquisition Regulation (FAR), plan the solicitation, selection and award of contracted efforts based on the selected strategies in the DBC. Obtain approvals to contract from appropriate authorities. As appropriate, execute the solicitation and selection of support and system contractors for the subsequent stages.

3.2.9 Create Documentation and Deliverables

The following work products are initiated or updated during this stage.

3.2.9.1 Draft Business Case (DBC)

The Draft Business Case (DBC) initiated during the Concept Approval Stage, is finalized during this stage to further identify the scope of the system. It should contain the high-level requirements, benefits, business assumptions, and program costs and schedules. It should also contain the information types, sensitivity, and security categorization. It records management decisions on the envisioned system early in its development and provides guidance on its achievement.

Each Level of IT Investment requires a Business Case documented in either on OMB Exhibit 300 or OMB Exhibit 53 Information. The appropriate OMB Exhibit documents the justification for the proposed investment in satisfying the particular business need, summarizing the need, the alternatives, the proposed solution, resources, schedules, risks, and other information. The OMB Exhibit, supported by information in SDLC Products provides the basis on which the investment is approved for inclusion in an investment portfolio via the CPIC Select process. An example and instructions for each submission are included in the CPIC Guide.

The Draft Business Case is adaptable as an OMB Exhibit, the OMB Exhibit 300 or the OMB Exhibit 53, depending on the project's IT investment level, estimated during this stage. GSA uses an automated portfolio management system known as eCPIC to document all IT investments. The business cases also serve as the basis for preparing the OMB Exhibit 300 and OMB Exhibit 53. The OMB Exhibits:

- Document the justification for the proposed investment
- Explain how an investment satisfies a particular business need, and
- Summarize the need, the alternatives, the proposed solution, resources, schedules, risks and other information.

The OMB Exhibits, supported by information in the SDLC products, provide the basis for approval of the project for inclusion in the FAS IT investment portfolio and its enrollment in the CPIC Select process. The format, guidelines, and templates for OMB Exhibit 300 and Exhibit 53 are located in the GSA CPIC Guide and in OMB Circular A-11. Work begins on OMB Exhibits during this stage.

[Appendix C-1](#) provides a template for the Draft Business Case. For smaller projects, the following work products can be part of the DBC.

Cost-Benefit Analysis

The Cost Benefit Analysis (CBA) is initiated during this stage and provides cost or benefit information for analyzing and evaluating alternative solutions to a problem and for making decisions about initiating, as well as continuing, the development of information technology systems. The analysis should clearly indicate the cost to conform to the security standards in the Technical Reference Model (TRM) and appropriate Federal guidance. [Appendix C-2](#) provides a template for the Cost-Benefit Analysis.

Feasibility Study

The Feasibility Study, initiated during this stage, provides an overview of a business requirement or opportunity and determines if feasible solutions exist before full life-cycle resources are committed. Software and hardware alternatives are reviewed and used to formulate preliminary platform options. Project feasibility leads to a "go" or "no go" decision about the project. Determining project feasibility is an interactive process of collecting and analyzing data and searching for cost-effective, viable technical solutions.

Use the project objectives, scope, and high-level requirements as the basis for determining project feasibility. Work with the business lines and user community to address technical issues and risks. Conduct research and investigate documents and other resources. [Appendix C-3](#) provides a template for the Feasibility Study.

3.2.9.2 Security Risk Assessment Report

Analyze threats to and vulnerabilities of the proposed system and the risk to the security of the overall system(s) to which it is a part or which it interacts.

The purpose of the security risk assessment is to analyze threats to and vulnerabilities of a system to determine the risks to the confidentiality, integrity, and availability of the system throughout the system development lifecycle. This is used as a basis for identifying appropriate and cost-effective security measures. Analyze the risk to the security of the overall system(s) to which it is a part or which it interacts. . Define the threat environment in which the system will operate. Document any vulnerabilities or weaknesses in the POA&M. Contact the FAS ISSM for guidance and support on the Security Risk Assessment.

3.2.9.3 System Security Plan

A formal plan detailing the types of computer security is required for the new system based on the type of information being processed and the degree of sensitivity. Those systems that contain, store, and reference personal information will be more closely safeguarded than most. Using the system categorization that was determined during the planning stage, and results from prior Security Risk Assessments; use NIST 800-53 (as amended), Annex 1, 2, or 3 to review and select the recommended security controls based on the type of system and function. Ensure the selected security controls, (planned or in place) are fully designed, documented and detailed in the security plan. The security plan should also provide a complete characterization or description of the information system, as well as attachments or references to key documents supporting the system's information security program (e.g., contingency plan, incident response plan, security awareness and training plan, rules of behavior, risk assessment, security assessment results, system interconnection agreements, security authorizations/accreditations, and plan of action and milestones).

3.2.9.4 Acquisition Plan

This document shows how all government human resources, contractor support services, hardware, software and telecommunications capabilities are acquired during the life of the project. The plan is developed to help ensure that needed resources can be obtained and are available when needed. GSA has implemented an automated tool, the Acquisition Planning Wizard that facilitates the preparation of the Acquisition Plan.

3.2.9.5 Project Management Plan

The Project Management Plan (PMP) is prepared for all projects, regardless of size or scope. It documents the project scope, tasks, schedule, allocated resources, and interrelationships with other projects. It describes the SDLC tailoring approach used to apply the SDLC to the project life cycle and a justification for any tailoring applied. The plan provides details on the functional units involved, required job tasks, cost and schedule performance measurement, milestone and review scheduling. Revisions to the Project Management Plan occur at the end of each stage and as information becomes available. The Project Management Plan should address the management oversight activities of the project. See Appendix C-24 for Project Management Plan Outline.

For smaller projects and Operations & Maintenance projects, the project can create a PMP-Lite instead of the larger scale PMP. In order to create a PMP-Lite document, a higher level Project Management Plan must exist at the Program Level which defines the information not contained within the PMP-Lite (e.g., Management Plans and Processes, Technical Processes). See Appendix C-25 for the PMP-Lite Outline.

The following plans may either be part of the PMP or a stand-alone document. Additionally, for larger programs, a program level plan may be established with each project documenting the project unique information as part of the plan.

Risk Management Plan

Complete the Risk Management Plan during this stage, identifying project risks and specifying the plans to reduce or mitigate the risks. [Appendix C-4](#) provides a template for the Risk Management Plan.

Configuration Management Plan

The Configuration Management (CM) Plan describes the process to identify, manage, control, and audit the project's configuration. The plan should also define the configuration management structure, roles, and responsibilities to be used in executing these processes. [Appendix C-5](#) provides a template for the Configuration Management Plan.

Quality Assurance Plan

The Quality Assurance (QA) Plan documents that the delivered products satisfy contractual agreements, meet or exceed quality standards, and comply with the project defined processes, which have been approved as tailored instances of this SDLC. [Appendix C-6](#) provides a template for the Quality Assurance Plan.

Systems Engineering Management Plan

The SEMP describes the system engineering process to be applied to the project; assigns specific organizational responsibilities for the technical effort, and references technical processes to be applied to the effort. Information that should be included in the SEMP is shown in [Appendix C-8](#).

3.2.9.6 Concept of Operations

The CONOPS is a high-level requirements document that provides a mechanism for users to describe their expectations from the system. Its content is based and traceable to the content of the Draft Business Case. Information that should be included in the CONOPS document is shown in Appendix C-7.

3.2.9.7 FAS Earned Value Management (EVM) Package

All projects that meet the FAS OCIO threshold for an Earned Value Management (EVM) threshold must prepare an EVM Approval package during the planning phase. The *FAS OCIO EVM Handbook* defines the format of the EVM Package and supporting process from a Project Manager perspective. For detailed information on GSA EVM Process, see the GSA EVMS Policy² for supporting information.

3.2.10 Hold Stage Review Activity

Upon completion of all Planning Stage tasks and commitment of resources for the next stage, the Project Lead, together with the project team should prepare and present a stage review activity with the Program Manager, Program Sponsor, and key project stakeholders. The review should address:

- I). Planning Stage activities and work product status
- II). Planning status for all subsequent life cycle stages (with significant detail on the next stage)
- III). Resource availability status
- IV). Risk assessments of subsequent life cycle stages

The Program Sponsor is charged with the decision to proceed to the next stage, iterate the planning tasks, or to terminate the project.

² GSA Acquisition Letter V-05-01, Implementation of Earned Value Management System (EVMS) Policy in GSA, Supplement 1.

4.0 REQUIREMENTS ANALYSIS STAGE

4.1 Overview

4.1.1 Objectives

The primary goal of this stage is to develop a basis of mutual understanding between the system owner/users and the project team about the requirements for the system. The result of this understanding is an approved requirements document that becomes the initial baseline for system design and a reference for determining whether the completed system performs as the system owner requested and expected.

The Requirements Analysis Stage begins when the previous stage documentation has been approved or by management direction. During this stage, the system is defined in more detail with regard to system inputs, processes, outputs, and external interfaces. This definition process occurs at the functional level. The system is described in terms of the functions to be performed, not in terms of computer programs, files, and data streams. The emphasis in this stage is on determining what functions must be performed rather than how to perform those functions.

Multiple instances of the Requirements Analysis Stage may occur in the project life cycle, especially if the project strategy includes life cycle segments for prototypes, pilots, or incremental builds/releases—each of which is allocated a subset of project requirements. In these cases, each such segment will revisit the requirements analysis work products developed in prior instances to update/extend them as appropriate to the requirements that are allocated to the specific segment.

4.1.2 Inputs

- Draft Business Case (DBC)
- Project Management Plan (PMP)
- Concept of Operations (CONOPS)
- System Security Plan

4.1.3 High Level Activities

- I). Define Requirements
- II). Create Documentation and Deliverables
- III). Conduct System Requirements Review
- IV). Establish Functional Baseline
- V). Hold Stage Review Activity

4.1.4 Work Products

- Functional Requirements Document
- Requirements Traceability Matrix
- Interface Control Document
- Test Plan
- Privacy Act Notice/Privacy Impact Assessment
- Updated work products from previous stages

4.2 Tasks and Activities

The following tasks are performed during the Requirements Analysis Stage. The tasks and activities actually performed depend on the nature of the project.

4.2.1 Define Requirements

The goals for defining requirements are to identify what functions are to be performed on what data, to produce what results, and for whom. The requirements must focus on the capabilities that are needed and the functions that are to be performed. Design issues and specifications must not be defined as part of the requirements. Generally, a requirement specifies an externally visible function or attribute of a system (i.e., a “what”) whereas as the design describes a particular instance of how that visible function will be achieved (i.e., the “how to”).

Requirements should be specified as completely and thoroughly as possible and contain the following attributes:

- Necessary – Absolute requirements that are to be verified are identified by “must” or “shall”. Goals or intended functionality are indicated by “will”.
- Correct – Each requirement is an accurate description of a feature or process of the product
- Unambiguous – Each requirement has only one interpretation
- Complete – Each requirement describes one result that must be achieved by the product
- Consistent – Individual requirements are not in conflict with each other
- Verifiable – Each requirement is stated in concrete terms and measurable quantities
- Traceable – The origin of each requirement is clear and can be tracked throughout the life cycle via the requirements traceability matrix (RTM)

In major projects, when alternative solutions must be examined or risk areas must be explored before system requirements can be finalized, pilot or prototype mini-projects may be required. These become life cycle segments in which exploration of the specific solution features or risk issues become the “requirements”. Once issues are resolved, risks are mitigated, and questions are answered, then the system requirements can be finalized and the primary SDLC can continue.

4.2.1.1 Define User Interface Requirements

The user interface requirements should describe how the user will access and interact with the system and how information will flow between the user and the system. The following are some of the issues that should be considered when defining these requirements:

- The users’ requirements for screen elements, navigation, reports, and help information
- Any applicable standards (e.g., Federal Government (508), GSA/FAS, industry) that apply to user interfaces
- The types of users who will access the system and for what purpose

4.2.1.2 Define Security Requirements

Define the security requirements in conjunction with the ISSO and other stakeholders who provide input into the system security area. This involvement affords complete definition of the security requirements for the system.

Implement applicable security procedures to assure data integrity and protection from unauthorized disclosure throughout the systems’ life cycle.

4.2.1.3 Define FOIA/Privacy Act Requirements

The collection, use, maintenance, and dissemination of information on individuals by any system in FAS will require a thorough analysis of both legal and privacy policy issues. Whether a system is automated or manual, privacy protections must be integrated into the development of the system. To ensure that FAS properly addresses the privacy concerns of individuals as systems are developed, FAS policy mandates that all IT initiatives develop and utilize the Privacy Impact Assessment (PIA) processes.

Templates for the Privacy Act Notice and the Privacy Impact Assessment, and directions for their preparation, are found in the applicable U.S. Code.³ For GSA's PIA policy, responsibilities, and procedures, see [GSA Order CPO 1878.2, Conducting Privacy Impact Assessments \(PIAs\) in GSA](#).

4.2.1.4 Define Data Requirements

Data requirements identify the data elements and logical data groupings that will be accessed (either input or output) by the system. The identification and grouping of data begins during the Requirements Analysis stage and is expanded in subsequent stages as more information about the data is known.

4.2.1.5 Define Other Requirements

Include all possible requirements including those for:

- Input and output requirements
- Performance
- Qualification requirements
- Architecture requirements (e.g, Hardware, Software, Communications, Network, etc.)
- Safety specifications, including those related to methods of operation and maintenance, environmental influences, and personnel injury
- Installation and acceptance requirements of the delivered software product at the operation and maintenance site(s)
- User operation and execution requirements
- User maintenance requirements

4.2.2 Create Documentation and Deliverables

4.2.2.1 Develop Requirements Traceability Matrix

A requirements traceability matrix (RTM) is a tool used to trace life cycle activities and work products to the requirements. The matrix establishes a thread that traces requirements from identification through final test and implementation.

All work products developed during the design, code, and testing processes in subsequent life cycle stages must be traced back to the requirements described in the FRD. This traceability assures that the system will satisfy all of the requirements and remain within the project scope.

Appendix C-28 provides a template for the Requirements Traceability Matrix (RTM).

4.2.2.2 Develop Functional Requirements Document

All of the requirements are documented in the Functional Requirements Document (FRD). The FRD describes the inputs to be supplied by the user or other sources, the processing that needs to occur and the outputs desired by the user or required by interfacing systems. The emphasis should be placed on specifying system functions without implying how the system will provide those functions. The FRD is created by integrating all of the requirements developed during this stage.

The FRD serves as the foundation for system design and development. This document captures functional/user requirements to be implemented in a new or enhanced system. These are complete, user-oriented functional and data requirements for the system that, through definition, analyses, and documentation, serve to ensure that user and system requirements have been collected and documented.

[Appendix C-9](#) provides a template for the Functional Requirements Document.

³ Privacy Act, 5 U.S.C. 552a

4.2.2.3 Develop an Interface Control Document

The project team responsible for the development of this system needs to articulate the other systems (if any) this system will interface with. Identify any interfaces and the exchange of data or functionality that occurs. All areas that connect need to be documented for security as well as information flow purposes.

The Interface Control Document (ICD) specifies the interface requirements imposed on one or more systems, subsystems, configuration items, or other system components to achieve one or more interfaces among these entities. Overall, an ICD can cover requirements for any number of interfaces between and among any number of systems. [Appendix C-11](#) provides a template for the Interface Control Document. For smaller systems with few interfaces, the ICD can be part of the FRD.

4.2.2.4 Develop Test Criteria and Plans

The Test Plan establishes the testing necessary to validate that the requirements have been met. It also ensures that a systematic approach to testing is established and that the testing is adequate to verify the functionality of the system.

The Test Plan includes the resources, roles and responsibilities, traceability back to the requirements, and techniques needed to plan, develop, and implement the testing activities that will occur throughout the life cycle. In this stage, the plan is written at a high level and focuses on identifying the testing phases and techniques. During the remaining stages of the life cycle, detailed information about the test plans and procedures is added to this Test Plan.

The types of test activities discussed in the subsequent sections identify more specifically the Integration and Test Stage of the life cycle that are included in the test plan and test analysis report.

- Unit/Module Testing
- Integration Testing
- Independent Security Testing
- Functional Qualification Testing
- User Acceptance Testing
- Beta Testing

Appendix C-10 provides a template for the Test Plan.

4.2.3 Conduct System Requirements Review

The System Requirements Review is conducted in the Requirements Analysis Stage and approved by the appropriate stakeholders (e.g., System Owner, System Users, Security, etc.). This is where the functional requirements identified in the FRD/RTM are reviewed to see if they are sufficiently detailed and are testable. It also provides the Project Lead with the opportunity to ensure a complete understanding of the requirements and that the documented requirements can support a detailed design of the proposed system.

4.2.4 Establish Functional Baseline

During the requirements analysis stage, the functional baseline, sometimes called a system requirements baseline, is established. The system requirements are baselined after the system owner's formal approval of the FRD. Once the requirements are baselined, any changes to the requirements must be managed according to the change control procedures established in the CM Plan.

4.2.5 Hold Stage Review Activity

Upon completion of all Requirements Analysis Stage tasks and commitment of resources for the next stage, the Project Lead, together with the project team should prepare and present a stage review activity with the Program Manager, Program Sponsor, and key project stakeholders. The review should address:

- I). Requirements Analysis Stage activities and work product status
- II). Planning status for all subsequent life cycle stages (with significant detail on the next stage)

- III). Resource availability status
- IV). Risk assessments of subsequent life cycle stages

The Program Sponsor is charged with the decision to proceed to the next stage, iterate the requirements analysis tasks, or to terminate the project.

5.0 DESIGN STAGE

5.1 Overview

5.1.1 Objectives

The objective of the Design Stage is to transform the requirements documented in the Functional Requirements Document into comprehensive high level and detailed designs sufficient to guide the work of the Development Stage. The decisions made in this stage address, in detail, how the system will meet the defined functional, physical, interface, security, and data requirements.

The goal of this stage is to define and document the functions of the system to the extent necessary to build the system to obtain the critical stakeholders understanding and approval. Prototyping of system functions can be helpful in communicating the design specifications.

Design Stage activities may be conducted in an iterative fashion, producing first a general system design that emphasizes the functional features of the system, then a more detailed system design that expands the general design by providing all the technical detail.

5.1.2 Inputs

- Functional Requirements Document
- Test Plan
- Interface Control Document
- Requirements Traceability Matrix

5.1.3 High Level Activities

- I). Determine System Structure
- II). Define the Application & Development Environments
- III). Design the System
- IV). Conduct Preliminary Design Review
- V). Create Documentation and Deliverables
- VI). Conduct Critical Design Review
- VII). Hold Stage Review Activity

5.1.4 Work Products

- System Design Document
- Implementation Plan
- Maintenance Manual
- Operations Manual/System Administration Manual
- Training Plan
- IT Contingency Plan
- Revised Previous Documentation

5.2 Tasks and Activities

Select the tasks to perform during the Design Phase Tasks using the nature of the project as the selection criterion. There are additional guidelines for selection and inclusion of tasks for in the Design Stage in Chapter 11, Alternate SDLC Work Patterns.

5.2.1 Determine System Structure

System Decomposition is an approach that divides the system into different levels of abstraction. Decomposition is an iterative process that continues until single purpose components (e.g., design entities or objects) can be identified. Decomposition is used to understand how the system will be structured, and the purpose and function of each entity or object.

The goal of the decomposition is to create a highly cohesive, loosely coupled, and readily adaptable design. A design is highly cohesive if each design entity in the system is essential for that unit to achieve its purpose. A loosely coupled design is composed of program units that are independent or almost independent.

5.2.2 Define the Application & Development Environments

Identify/specify the Target operating environment. How and where the application will reside. Assign responsibility for this activity.

Identify/specify the Development environments for design, development, testing, and implementation. How and where the application will be developed, tested, and implemented. Assign responsibility for this activity.

5.2.3 Design the System

The initial focus of IT system design is to define the general characteristics of the system. Design the data storage and access for the database layer. Design the User Interface at the desktop layer. Include the application logic design in the Business Rules layer.

Establish a top-level architecture of the system and document it. The architecture identifies items of hardware, software, and manual-operations. Allocate all the system requirements among the hardware configuration items, software configuration items, and manual operations. For major systems, the Project Lead should convene a Preliminary Design Review to assess the high-level design before proceeding to the Detailed Design.

For each software configuration item, transform the allocated requirements into an architecture that describes its top-level structure and identifies its software components. Ensure that allocation of all the requirements for the software configuration item occurs in the software components and further refine these to facilitate detailed design of each component.

Develop and document a top-level design for the interfaces external to the software item and between the software components of the software item.

5.2.3.1 Design User Interface

One of the first steps is to design a user interface that is appropriate for the users, content, and operating environment for the system. The user interface design is defined for all categories of users and for all of the functions they will perform. Prototyping is a valuable tool that can be used to present the user interface design to the users so that design weaknesses can be identified and resolved early. Prototypes can also help to gain user acceptance of the interface.

As part of designing the user interface, the following areas are defined:

- Menu and screen hierarchy
- Screen navigation
- Data entry screens
- Report screens
- Online help
- Error messages

5.2.3.2 Design System Interfaces

During this stage, the design for how the system will interface with other systems as defined by the Interface Control Document and the FRD. The various interface designs should be submitted to the appropriate system owner for review to verify that the interface design is consistent with their design and understanding of the requirements. This helps to ensure that any incompatibilities with the interfaces are identified early in the design process and corrective actions can be initiated to assure each interface is properly designed and coded.

5.2.3.3 Design System Security Controls

During the design stage, the developer allocates the security requirements to the specific modules with the design for enforcement purposes. The National Institute of Standards and Technology (NIST) Special Publication 800-53, Recommended Security Controls for Federal Information Systems defines these requirements in detail.

Detailed security requirements provide users and administrators with instructions on how to operate and maintain the system securely. They should address all applicable computer and telecommunications security requirements, including:

- System access controls
- Marking, handling, and disposing of magnetic media and hard copies
- Computer room access
- Account creation, access, protection, and capabilities; operational procedures
- Audit trail requirements
- Configuration management
- Processing area security
- Employee check-out
- Disaster recovery emergency procedures

5.2.3.4 Build Logical Data Model

The logical data model defines the flow of data through the system and determines a logically consistent structure for the system. The logical data model is a representation of a collection of data objects and the relationships among these objects. The data model is used to provide the following functions:

- Transform the business entities into data entities
- Transform the business rules into data relationships
- Resolve the many-to-many relationships
- Determine a unique identifier for each data entity
- Define the attributes for each data entity
- Define the data integrity rules

5.2.3.5 Define/Confirm System Architecture

During the design stage, the system architecture is either defined or confirmed for the system. If the system will be operating in a new environment, define the architecture that is the best, cost effective solution that satisfies the requirements. If the system will be operating in an existing environment, confirm that this environment will support that additional requirements levied (e.g., hardware, communications, performance, availability, etc.) on this environment by the system.

5.2.3.6 Design Physical Model and Database Structure

The physical model is a description of the dynamics, data transformation, and data storage requirements for the system. The physical model maps the logical model to a specific technical reality.

At this point, complete the data dictionary with all information on data elements, entities, files, physical, and data conversion requirements.

5.2.3.7 Update the Requirements Traceability Matrix

During system design, the allocation of requirements to configuration items and then to components of the configuration items is documented in the RTM by adding columns containing references to configuration items and components. This will establish a clear relationship between components and requirements for use during subsequent verification/validation activities and during life cycle maintenance.

5.2.4 Conduct Preliminary Design Review

This is an iterative review of the system design as it evolves through the Design Stage. This review determines whether the initial design concept is consistent with the overall architecture and satisfies the functional, security, and technical requirements in the Functional Requirements Document.

5.2.5 Create Documentation and Deliverables

5.2.5.1 Develop Conversion/Migration/Transition Strategies

When defining the requirements and design the plans for converting, migrating, and transitioning current information to the new system, it is important to be particularly conscious of plan design if conversion means re-engineering existing processes.

The Conversion Plan describes the strategies involved in converting data from an existing system to another hardware or software environment. It is appropriate to re-examine the original system's functional requirements for the condition of the system before conversion to determine if the original requirements are still valid. [Appendix C-12](#) provides a template for the Conversion Plan

5.2.5.2 Develop System Design Document

The System Design Document (SDD) describes the following:

- system requirements
- operating environment
- system and subsystem architecture
- files and database design
- input formats and output layouts
- user interface
- detailed design
- processing logic, and
- external interfaces.

The System Design Document, in conjunction with the Functional Requirements Document (FRD) [complete at this stage], provides general and system design specifications for the system and reflects the technological perspective of the system design. It includes all information required for the review and approval of the project development. The sections and subsections of the design document may be organized, rearranged, or repeated as necessary to reflect the best organization for a particular project. [Appendix C-13](#) provides a template for the System Design Document.

5.2.5.3 Develop Implementation Plan

The Implementation Plan describes system deployment and implementation in the operational environment. The plan contains an overview of the system, a brief description of the major tasks involved in the implementation, the overall resources needed to support the implementation effort (such as hardware, software, facilities, materials, and personnel), and any site-specific implementation requirements. Updates to the plan may occur during the Development Stage; the final version will be available in the Integration and Test Stage for guidance during the Implementation Stage. [Appendix C-14](#) provides a template for the Implementation Plan.

5.2.5.4 Develop Maintenance Manual

The Maintenance Manual provides maintenance personnel with the information necessary to maintain the system effectively. The manual provides the definition of the software support environment, the roles and

responsibilities of maintenance personnel, and the regular activities essential to the support and maintenance of program modules, job streams, and database structures. The Maintenance Manual may also provide additional information to facilitate the maintenance and modification of the system. It may also include Appendices to document various maintenance procedures, standards, or other essential information. [Appendix C-15](#) provides a template for the Maintenance Manual.

5.2.5.5 Develop Operations Manual/System Administration Manual

Develop the Operations Manual and/or the System Administration Manual, as determined during the Planning Stage. Identify required Security operational procedures. These procedures will become the Rules of Behavior to be included in the System Security Plan.

For mainframe systems, the Operations Manual provides computer control personnel and computer operators with a detailed operational description of the information system and its associated environments, such as machine room operations and procedures. The Systems Administration Manual serves the purpose of an Operations Manual in distributed (client/server) applications. [Appendix C-16](#) provides a template for the Operations Manual and [Appendix C-17](#) provides a template for the Systems Administration Manual. The Project Lead must determine, based on consultation with user and operations stakeholders, which of these documents or other such documents is required.

5.2.5.6 Design User Training

The Training Plan (TP) outlines the objectives, needs, strategy, and curriculum for training users on the new or enhanced information system. The plan presents the activities needed to support the development of training materials, coordination of training schedules, reservation of personnel and facilities, planning for training needs, and other training-related tasks. The Training Plan includes the target audience and topics on the list of training needs. It includes, in the training strategy, the methodology and the format of the training program, the list of topics to be covered, materials, time, space requirements, and proposed schedules. [Appendix C-18](#) provides a template for the Training Plan.

5.2.5.7 Develop User Manual

The User Manual contains all essential information for the user to make full use of the information system. This manual includes a description of the system functions and capabilities, contingencies and alternate modes of operation, and step-by-step procedures for system access and use. [Appendix C-19](#) provides a template for the User Manual.

5.2.5.8 Develop IT Contingency Plan

The Contingency Plan contains emergency response procedures; backup arrangements, procedures, and responsibilities; and post-disaster recovery procedures and responsibilities. Contingency planning is essential to ensure that FAS systems are able to recover from processing disruptions in the event of localized emergencies or large-scale disasters. It is an emergency response plan; developed in conjunction with application owners and maintained at the primary and backup computer installation to ensure continuity of support should events occur that could prevent normal operations. Regularly review, update, and test the Contingency Plan to ensure that restoration of vital operations and resources occurs as quickly as possible and to keep system downtime to an absolute minimum. A Contingency Plan is complementary to a disaster recovery plan, business continuity plan, and an emergency plan. If the system/subsystem is to be located within a facility with an acceptable contingency plan, add system-unique contingency requirements as an annex to the existing facility contingency plan. [Appendix C-20](#) provides a template for the Contingency Plan.

5.2.6 Conduct Critical Design Review

The Critical Design Review (CDR) is a formal technical review of the system design and is held with the key stakeholders (e.g., System owner, users, security, hosting, other system owners, etc.). The purpose of the review is to demonstrate that the system design addresses all functional, security, and technical requirements and is consistent with the overall architecture.

5.2.7 Hold Stage Review Activity

Upon completion of all Design Stage tasks and commitment of resources for the next stage, the Project Lead, together with the project team should prepare and present a stage review activity with the Program Manager, Program Sponsor, and key project stakeholders. The review should address:

- I). Design Stage activities and work product status
- II). Planning status for all subsequent life cycle stages (with significant detail on the next stage)
- III). Resource availability status
- IV). Risk assessments of subsequent life cycle stages

The Program Sponsor is charged with the decision to proceed to the next stage, iterate the design stage tasks, or to terminate the project.

6.0 DEVELOPMENT/INTEGRATION & TEST STAGE

6.1 Overview

6.1.1 Objectives

The objective of the Development/Integration & Test Stage will be to convert the work products of the Design Stage into tested components of a complete information system and prove that the developed system satisfies the requirements defined in the FRD. Therefore, it is critical that a complete Project Management Plan, System Design Document, and Test Plan are in place before beginning this stage. This stage is sufficiently robust to integrate and perform required test and verification activities against the entire system. Although much of the activity in the Development/Integration & Test Stage addresses the elements of software that make up the system, this stage also puts in place development and test environments sufficient to perform the tests and verification activities of the software components developed. The hardware, software, and communications elements of the entire system are assembled or simulated in order to provide a test bed for system requirements.

The activities of this stage translate the system design produced in the Design Stage into a set of validated configuration items representing all information system requirements. The Development/Integration and Test stage contains activities for building, testing, and qualifying individual configuration items against their allocated requirements. For a complex system design, with multiple configuration items, there may be several instances of the Development/Integration and Test Stage in the project life cycle, each stage dealing with an individual configuration item. Integration and test activities are performed first in a Development environment, which may not be as robust as the Integration and Test or Implementation environments. When all instances of the development are complete, all configuration items that comprise the system will be compiled and transferred into the Test environment for the integration and test.

If a development organization is responsible for multiple configuration items, then a single instance of the Development/Integration and Test Stage may include the development, integration, and qualification testing of the set of such items. Both the acquisition plan and the system design affect the number of development threads in a single Integration and Test activity.

For complex systems, it is advisable to integrate configuration items incrementally rather than all at once, allowing a staggered development schedule. This stage results in a fully integrated and qualified system, insofar as verifiable, in an Integration and Test environment, which may simulate applicable characteristics of the operating environment. This stage includes several types of tests:

- With the support of developers, verify the ability to build and integrate components into delivered configuration items within the Integration and Test environment.
- Test the integration of configuration items into the target system. The development team analyzes the results Integration tests to ensure that the subsystems integrate properly into the target system.
- Next, the testing team conducts and evaluates system tests to ensure the target system meets all technical requirements, including performance requirements.
- Developers are encouraged to use Static Code Analysis tools during the development process to identify vulnerabilities as code is developed. Then, the testing team and the Security Program Manager perform security tests to validate that the system meets access and data security requirements.
- Finally, users participate in acceptance testing to confirm that the developed system meets all user requirements as stated in the FRD. Execute User Acceptance with the users employing simulated or real target platforms and infrastructures.

To the extent that the Integration and Test environment simulates the operating environment, this stage readies the system for Implementation. The Implementation Stage that follows will further verify system performance in an Implementation environment that is closer to the operating environment.

6.1.2 Inputs

- Project Management Plan
- System Design Document
- Unit and Integration Test Plans

6.1.3 High Level Activities

- I). Establish the Development and Test Environments
- II). Perform Development
- III). Integrate Software
- IV). Install Software
- V). Develop/Integrate System Security Controls
- VI). Establish the Test Environments
- VII). Conduct Integration Test
- VIII). Hold Test Readiness Reviews
- IX). Conduct Subsystem/System Testing
- X). Conduct Security Vulnerability Testing
- XI). Conduct Acceptance Testing
- XII). Create Documentation and Deliverables
- XIII). Hold Stage Review Activity

6.1.4 Work Products

- Software Development Document
- System Software
- Test Files/Data
- Test Analysis Report
- Problem Report
- Revised Previous Documentation

6.2 Tasks and Activities

The tasks and activities actually performed depend on the nature of the project. The following tasks should be completed during the Development/Integration and Test stage.

6.2.1 Establish the Development and Test Environments

Establish the various development teams and ensure the development environment is ready.

6.2.2 Perform Development

6.2.2.1 Develop the Software

For the software configuration item, code each software component in accordance with the System Design Document and established standards.

6.2.2.2 Develop the Database

For the database, implement the database as defined by the Database Design Document. If conversion of an existing system or data is necessary, perform a preliminary run of the process described in the Conversion Plan.

6.2.3 Conduct Unit Test

Unit testing verified the inputs and outputs for each model. Successful unit testing indicates the validity of the functions (i.e., requirements) performed by the module and shows traceability to the design. During unit testing, each module is tested individually and the module interface is verified for consistency with the design specification. All important processing paths through the module are tested for expected results as well as all error handling paths.

Unit testing is driven by test cases and test data that are designed to verify requirements, and to exercise all program functions, edits, in-bound and out-of-bound values, and error conditions.

6.2.4 Integrate Software

Successively integrate and test the software components until the software configuration item has been integrated and tested in accordance with the integration plan and satisfies its allocated requirements.

6.2.5 Install Software

Demonstrate the installation of the software configuration item in the Integration and Test environment. The resources and information necessary to install the software product is determined and available. The developer of the software configuration item assists the Integration and Test organization with the installation activities and provides support for the Integration and Test of the software configuration item during System Integration and Test. This support includes correction of any deficiencies found in the software configuration item, including in prior stage work products, during conduct of Integration and Test. Development support continues through the Implementation stage and into Operations & Maintenance, until established acceptance criteria have been satisfied.

6.2.6 Develop/Integrate System Security Controls

The security controls described in the System Security Plans are developed, integrated, configured and tested in the new information systems architecture. If applicable, the Security Plans may also be revised during this stage to account for any adjustments or modifications that came about during integration or testing of the security controls. The Security Plans for information systems already in operation may call for the development/integration of additional security controls to supplement the controls already in place or the modification of selected controls that are deemed to be less than effective. For these operational systems, a Risk Assessment is required to determine the effect of the changes on the overall security posture of the system. Based on the results of the Risk Assessment, a new C&A may be required.

6.2.7 Establish the Test Environments

Establish the various test teams and ensure the appropriate test environments are ready.

6.2.8 Conduct Integration Test

During integration test, the individual components are successively integrated together and tested in a systematic manner in accordance with the Integration Test Plan. An incremental approach to integration enables verification that, as each new component is integrated, it continues to function as designed and both the component and the integrated product satisfy their allocated requirements.

Integration testing is a formal procedure that must be carefully planned and coordinated with the completion dates of the unit-tested modules. Integration testing begins with a structure where called sub-elements are simulated by stubs. A stub is a simplified program or dummy module designed to provide the response that would be provided by the real sub-element. A stub allows testing of calling program control and interface correctness. Stubs are replaced by unit-tested modules or builds as integration testing proceeds. This process continues one element at a time until the entire system has been integrated and testing.

Integration testing may be performed using “bottom up” or “top down” techniques. Most integration testing makes use of both techniques due to scheduling and other constraints.

6.2.9 Hold Test Readiness Reviews

A Test Readiness Review (TRR) is typically held before the beginning of System testing. However, one can also be held prior to the beginning of other test phases. The purpose of the TRR is to:

- Verify the results of the previous testing phase and that all known problems are documented
- Verify that all required work products are completed for the next phase of testing to include the detailed test plans, test cases, and test environments
- Verify that the test team has the required resources to properly execute the test

6.2.10 Conduct System Testing

During system testing, the completely integrated system is tested to validate that it meets all requirements. Either the project team or an independent test team conducts system testing to assure that the system performs as expected and that each function executes without error – under both normal and high-load conditions.

System testing is conducted in the system test environment using the methodology and test cases described in the System Test Plan. The system test environment should be as close to possible as the production environment. The results of each test are documented in the Test Analysis Report ([Appendix C-21](#)). Any failed components should be migrated back to the development stage for rework, and the passed components should be migrated ahead for security testing.

6.2.11 Conduct Security Testing

The test and evaluation team will again create or load the test database(s) and execute security (penetration) test(s) that will verify the confidentiality, integrity, and availability objectives of the information system and the data it stores, processes, and transmits. Depending on the security categorization of the information and/or the information system, the security test and evaluation may need to be conducted by an independent test and evaluation team. All tests will be documented, similar to those above. Failed components will be migrated back to the development stage for rework, and passed components will be migrated ahead for acceptance testing.

The IT Systems Security Certification & Accreditation activities must comply with the guidelines and policies established by the GSA/FAS Office of the CIO. Consult the FAS ISSO on any variance.

Issuance of an IT Systems Security Certification and Accreditation Memoranda will be in accordance with guidance contained in GSA Procedural Guide on Managing Enterprise Risk (CIO IT Security 06-30 R7 – or most recent version).

The Systems Security Plan and certification/accreditation package approval must be in place prior to implementation and must undergo renewal every three years thereafter.

6.2.12 Conduct Acceptance Testing

Acceptance of the delivered system is the ultimate objective of a systems development project. Acceptance testing is used to demonstrate the system's compliance with the system owner's requirements and acceptance criteria.

At the system owner's discretion, acceptance testing may be performed by the project team, by the system owner and users with support from the project team, or by an independent verification and validation team. To the extent possible, users should participate in acceptance testing to assure that the system meets the users' needs and expectations.

Acceptance testing is conducted in the production environment using acceptance test data and test procedures defined in the Acceptance Test Plan. Testing is designed to determine whether the system meets the functional, performance, and operational requirements. Acceptance testing usually covers the same requirements as the system test.

All tests will be documented, similar to those above. Failed components will be migrated back to the development stage for rework, and passed components will migrate ahead for implementation.

6.2.13 Create Documentation and Deliverables

6.2.13.1 Prepare Software Development Folder

This document contains all material information pertaining to the development of each component or configuration item, including the test cases, software, test results, approvals, and any other items that will help explain the functionality of the software. The Software Development Folder is project specific and defined by the PM.

6.2.13.2 Deliver System Software

This is the actual software developed for the software configuration item. It is used for the Integration and Test Stage and finalized before implementation of the system. Include all the disks (or other medium) used to store the information. Format is digital, under configuration control, and established for the specific project.

6.2.13.3 Deliver Test Files/Data

All the information used for software testing should be provided at the end of this stage. Provide the actual test data and files used. Format is digital, under configuration control, and established for the specific project.

6.2.13.4 Deliver Test Analysis Report

This report documents each test – unit/module, subsystem integration, system, user acceptance and security. [Appendix C-21](#) provides a template for the Test Analysis Report.

6.2.13.5 Deliver Problem Report

Document problems encountered during testing; the form is attached to the test analysis reports.

6.2.13.6 Deliver User Acceptance Test Report

This report documents the results of the user acceptance report including the summary of the test, detected defects, and final recommendations/action items. Appendix C-26 provides a template for the User Acceptance Test Report.

6.2.14 Hold Stage Review Activity

Upon completion of all Development/Integration & Test Stage tasks and commitment of resources for the next stage, the Project Lead, together with the project team should prepare and present a stage review activity with the Program Manager, Program Sponsor, and key project stakeholders. The review should address:

- I). Development/Integration & Test Stage activities and work product status
- II). Planning status for all subsequent life cycle stages (with significant detail on the next stage)
- III). Resource availability status
- IV). Risk assessments of subsequent life cycle stages

The Program Sponsor is charged with the decision to proceed to the next stage, perform additional testing, or to terminate the project.

7.0 IMPLEMENTATION STAGE

7.1 Overview

7.1.1 Objectives

The Implementation Stage commences after the user(s) accept the system in the Integration and Test environment and the TRB authorizes implementation. **Figure 8-1** depicts the Implementation Stage.

System implementation may occur in a single migratory action, in a series of discreet phases, in a beta test environment, in a parallel operating environment, or some hybrid of these modes. The Implementation environment may be more robust than the Integration and Test environment but still not identical to the target operational environment. The target operational environment, including infrastructure, personnel, and procedures, is prepared for system deployment. Once the system has received user acceptance and both Program Sponsor and review/approval authority give approval to deploy, the system moves into full operation. This stage in concludes with a Post-Implementation Review, completing both the Implementation Stage and the development project when the system transitions into the Operations and Maintenance stage.

There is limited Implementation Staging capacity within the FAS infrastructure. It may be necessary and operationally effective to perform multiple systems implementations simultaneously, both to optimize use of implementation bandwidth and to ensure that systems operate effectively together before full deployment. When close coupling exists among systems, project planning should address performance of detailed system and acceptance testing on them together during a combined Integration and Test activity before Implementation.

Tasks and activities in the implementation stage are associated with certain work products described in section 8.3. The tasks and activities actually performed depend on the nature of the project.

7.1.2 Inputs

- Software Development Document
- System Software

7.1.3 High Level Activities

- I). Conduct User Training
- II). Perform Data Entry or Conversion
- III). Establish the System-specific Implementation Environment
- IV). Install/Test the System Software in the Implementation Environment
- V). Conduct Baseline Security Vulnerability Scan
- VI). Transition and Evaluate the System in the Production Environment
- VII). Conduct Post-Implementation Review
- VIII). Create Documentation and Deliverables
- IX). Hold Stage Review Activity

7.1.4 Work Products

- Change Implementation Notice
- Delivered System
- Change Control Board Decision Document.
- Project Termination Plan

7.2 Tasks and Activities

The implementation notice is sent to all users and organizations affected by the implementation. Additionally, it is good policy to make internal organizations not directly affected by the implementation aware of the schedule so that allowances are in place for a disruption in the normal activities of that section. Some notification methods are email, internal memo to heads of Agency components, and voice tree messages. The notice should include:

- The schedule of the implementation
- A brief synopsis of the benefits of the new system
- The difference between the old and new system
- Responsibilities of end user affected by the implementation during this stage, and
- The process to obtain system support, including contact names and phone numbers.

7.2.1 Conduct User Training

User training is an important factor in the success of the operational system. During training, most users will receive their first hands-on experience with the system. The objective of training is to provide the user with the basic skills needed to effectively use the system and to raise the user's confidence and satisfaction with the product.

The type of training will depend on the complexity of the system, and the number and location of users to be trained. Alternative training formats include formal classroom training, one-on-one training, train the trainer training, computer-based training, and sophisticated help screens and online documentation. Conduct the training as described in the training plan.

7.2.2 Perform Data Entry or Conversion

When implementing a new system, there is often old data that migrate to the new system. This data can be in a manual or an automated form. Regardless of the format of the data, the tasks in this section are two fold, data input and data verification. When replacing a manual system, hard copy data will include manual data entry into the automated system. Manually entered data should undergo verification to ensure accuracy. This is also the case in data transfer, where data fields in the old system may contain erroneous or inconsistent entries that will affect the integrity of the new database. Verification of the old data is an imperative for implementing a useful computer system.

One way to accomplish system operation and data integrity is through parallel operations. Parallel operations consist of running the old process or system and the new system/process simultaneously until the new system is certified. In this way if the new system fails in any way, the operation can continue on the old system while the bugs are worked out while the new system undergoes corrective action.

7.2.3 Establish the System-specific Implementation Environment

Ensure that the Implementation Environment is ready for system installation and implementation testing.

7.2.4 Install/Test the System Software in the Implementation Environment

With developer support as required, install the system software and conduct System Tests and User Acceptance Tests as prescribed in the Implementation Plan. Document test results and any deficiencies in the Test Analysis Reports and Problem Reports, with deficiencies going back to the Development stage for correction.

Upon successful completion of the implementation tests, performance validation, and user acceptance, the TRB will make a determination of the system's readiness for moving into the Production Environment and issue guidance to the PMO's CCB to issue a preliminary Configuration Control Board (CCB) Decision Document authorizing the transition.

7.2.5 Evaluate the System in the Production Environment

Install the system software in the production environment for limited-use evaluation. As described in the Implementation Plan, evaluate and initiate its operation incrementally across an appropriate subset of users and infrastructure to demonstrate readiness for full operation.

7.2.6 Conduct Security C&A

Some types of security controls (e.g. management and operational controls) cannot be tested and evaluated until the information system is in its intended operational environment. Therefore, security controls developed/integrated for an information system must undergo Security certification after implementation and prior to commencing formal operations in the operational environment. The certification activities are designed to ensure the appropriate technical, management and operational controls are in place and working properly and effectively to assure the confidentiality, integrity, and availability of the information system and the data it stores, processes, and transmits. Depending on the security categorization of the information and/or the information system, the security certification activities may need to be conducted by an independent entity. After the certification activities are completed, the certification package is submitted to the ISSM who reviews it, certifies the system is working at an acceptable level of risk (if applicable), and forwards the certification statement to the DAA who accepts or rejects the residual risks. If the residual risks are accepted, the DAA issues an Authority to Operate (ATO) the system for a maximum of three years. If the level of risk is determined to be unacceptable, the Certification package is sent back to the PM for further risk mitigation actions until the residual risk is deemed acceptable by the DAA.

7.2.7 Authorize Transition to Full Operations

When the criteria for full implementation are satisfied, user acceptance has been achieved, and an ATO has been granted, the Project Lead will submit a request to the CCB for approval to promote the system into full operation. The CCB will evaluate the request and issue a final CCB Decision Document, establishing the Production Baseline, and authorizing full implementation, in accordance with the Implementation Plan. Observation of system performance in full operation and attendant user acceptance, in accordance with criteria established in the Implementation Plan, establish the basis for holding a Post-Implementation Review to formally conclude the Implementation Stage.

7.2.8 Conduct Post-Implementation Review

The review is conducted at the end of the Implementation Stage. A post-implementation review is conducted to ensure that the system functions as planned and expected; to verify that the system cost is within the estimated amount; and to verify that the intended benefits are derived as projected. Normally, this is a one-time review, and it occurs after a major implementation; it may also occur after a major enhancement to the system. The results of an unacceptable review are submitted to the Program Sponsor for its review and follow-up actions. The Program Sponsor may decide it will be necessary to return the deficient system to the responsible system development Project Lead for correction of deficiencies. The guidelines for the Post Implementation Review are provided in [Appendix C-22](#).

7.2.9 Create Documentation and Deliverables

7.2.9.1 Change Implementation Notice

A formal request and approval document for changes made during the Implementation Stage.

7.2.9.2 Delivered System

After the PM and System Sponsor sign the Implementation Stage Review and Approval Certification, the system, including the production version of the data repository, is formally delivered to the Operations Organization to commence the Operations and Maintenance Stage. Format is digital, under configuration control, and established for the specific project.

7.2.9.3 Change Control Board Decision Document.

Upon system acceptance and completion of implementation, the TRB will instruct the PMO to release the CCB Decision Document; this is a formal ratification of the readiness of the system to enter full operation and signals approval of all system products as the Production Baseline. The guidelines for the CCB Decision Document appear in the project Configuration Management Plan, Appendix C-5.

7.2.9.4 Project Termination Plan

The Agency, FAS Management Council, or System Sponsor may invoke the Project Termination Plan at any stage after Program Authorization should it become necessary to cancel the project. The PTP

provides the basis for direction and control of the technical and business aspects for completing project closeout. It provides the approval to finish any remaining project tasks and coordinates the activities with the involved Operational and Support Organizations to accomplish successful project closure. Create the PTP, as necessary, at any time after project authorization.

7.2.10 Hold Stage Review Activity

Upon completion of all Implementation Stage tasks and commitment of resources for the next stage, the Project Lead, together with the project team should prepare and present a stage review activity with the Program Manager, Program Sponsor, and key project stakeholders. The review should address:

- I). Implementation Stage activities and work product status
- II). Planning status for all subsequent life cycle stages (with significant detail on the next stage)
- III). Resource availability status
- IV). Risk assessments of subsequent life cycle stages

The Program Sponsor is charged with the decision to proceed to the next stage, perform additional testing, or to terminate the project.

8.0 OPERATIONS AND MAINTENANCE STAGE

8.1 Overview

8.1.1 Objectives

More than half of the life cycle costs are attributable to the operations and maintenance of systems. In this stage, the utilized system is scrutinized to ensure that it continues to meet the users' needs. Providing user support is an ongoing activity. New users will require training, problems are detected, and new requirements arise. The key considerations for this stage are: to meet users' needs; to ensure that the system continues to perform as specified in the operational environment; and that the system can logically adapt to the evolving business and user requirements. This kind of agenda may necessitate modification to the existing code, development of new code and/or changes to the hardware configuration. Changes will be required to fix problems, possibly add features and make improvements to the system. This stage will continue as long as the system is in use.

8.1.2 Inputs

- System Software
- System Change Requests
- System Documentation

8.1.3 High Level Activities

- I). Perform O&M Planning
- II). Perform O&M Requirements Analysis
- III). Perform O&M Design
- IV). Perform O&M Development/Integration & Test
- V). Perform O&M Implementation
- VI). Perform Routine Maintenance
- VII). Create Documentation and Deliverables
- VIII). Hold Stage Review Activity

8.1.4 Work Products

- Updated SDLC Documentation
- In Progress Review Reports
- User Satisfaction Review Report
- Security Vulnerability Scan

8.2 Tasks and Activities

8.2.1 Perform O&M Planning

For many of the systems within FAS OCIO, they have been on-going for several years (even decades) and the majority of work is associated with O&M. As part of the O&M activities, the various efforts are defined as projects – along with the activities described earlier in this SDLC. However, many of these activities are shortened to realistically address the schedule and costs constraints associated with O&M activities. The following sections describe these activities in more detail.

8.2.1.1 Identify System Change Requests

In this stage, system changes requests (SCRs) are identified, categorized, and assigned an initial priority ranking. Additionally, a proposed date/schedule for each SCR is defined. Each request for a modification is evaluated to determine its classification and handling priority. Various classifications are defined and typically include the following:

- Problem which is a change to the system in order to correct its defective behavior
- Enhancement which is a requested change to the system not originally defined in the requirements or supporting design
- Scheduled maintenance which is a change to the system after delivery to improve performance, maintainability, or perform routine maintenance on the system or supporting components/architecture

Priorities are defined and typically are defined as emergency, high, medium, or low.

8.2.1.2 Perform Release Planning

At the beginning of each release planning stage, the list of approved system changes are assigned to a particular release. As each release is planned out in further detail, the proposed system changes for each release are verified and the appropriate changes are made (e.g., delayed to a later release, moved to an earlier release, cancelled, etc.).

8.2.1.3 Update/Create Planning Documents

After the scope for the release is defined, a detailed project schedule is created along with a supporting Project Management Plan-Lite. These plans are communicated and approved by the appropriate stakeholders. See Appendix C-25 for Project Management Plan-Lite Outline.

8.2.2 Perform O&M Requirements Analysis

8.2.2.1 Analyze SCR Requirements

During this activity, each SCR is analyzed in detail. In identifying the requirements for the modification, examine all work products (e.g., requirements documents, design specifications, database documents, etc.) that are affected. Additionally, each requirement is analyzed for its impact on other requirements (e.g., interfaces, security, user interface, etc.).

8.2.2.2 Prepare Functional Requirements Document

All of the requirements for the release are documented in the FRD which may include a description of the release, SCR list and short description of each SCR, Interface Control Document, and an updated Requirements Traceability Matrix. The requirements should be reviewed and signed-off by the appropriate stakeholders. For larger releases, this may be done through a more formal Functional Requirements Review.

8.2.3 Perform O&M Design

8.2.3.1 Perform System Design

In the design stage, all current system and project documentation, existing software and databases, and the output of the O&M Requirements Analysis stage are used to design the modification to the system. As part of this, the user interface, supporting software, database, and system architecture are designed to accommodate the SCRs/requirements planned as part of this release.

8.2.3.2 Create/Update Documentation and Deliverables

The following documents are either created or updated from earlier project/system documentation:

- System Design Document
- Conversion Plan
- Implementation Plan
- IT Contingency Plan
- System Security Plan
- Test Plan

8.2.3.3 Conduct Critical/System Design Review

For major releases, a CDR may be held with the Project Sponsor, Project Manager, Project Team, and other key stakeholders. The format of the CDR may be influenced by several criteria including scope, complexity, risk, business value, and the audience.

8.2.4 Perform O&M Development/Integration & Test

8.2.4.1 Perform Coding and Unit Testing

Implement the changes into the code and perform unit testing.

8.2.4.2 Conduct Testing

For O&M releases, a system must undergo testing to ensure that is meeting all the requirements for the release and that the new implementation has not negatively affected the other parts of the system. Depending on the size and complexity of the release, some or all of the following testing may need to be performed.

8.2.4.3 Integrate and Test Software.

After the modifications are coded and unit tested, or at appropriate intervals during coding, the modified components are integrated with the system, and integration and regression tests are refined and performed.

8.2.4.4 Conduct System Test

During system testing, the completely integrated system is tested to validate that it meets all requirements. Either the project team or an independent test team conducts system testing to assure that the system performs as expected and that each function executes without error – under both normal and high-load conditions.

8.2.4.5 Conduct Security Test

The test and evaluation team will again create or load the test database(s) and execute security (penetration) test(s) that will verify the confidentiality, integrity, and availability objectives of the information system and the data it stores, processes, and transmits. Depending on the security categorization of the information and/or the information system, the security test and evaluation may need to be conducted by an independent test and evaluation team.

8.2.4.6 Conduct Acceptance Test

Acceptance of the delivered system is the ultimate objective of a systems development project. Acceptance testing is used to demonstrate the system's compliance with the system owner's requirements and acceptance criteria.

8.2.4.7 Create/Update Documentation and Deliverables

The following documents are either created or updated from earlier project/system documentation:

- Software Development Folder
- System Software
- Test Files/Data
- Test Analysis Reports
- Software Change Requests/Problems Reports
- Security Scanning Report

8.2.5 Perform O&M Implementation

The following activities may be performed as part of the implementation:

- Notify Users and Other Stakeholders of Pending Release

- Perform Data Conversion/Data Migration
- Install/Test the Release in the Implementation Environment
- Conduct Final Security Scan
- Create/Update Deliverables

8.2.6 Perform Routine Maintenance

Operations support is an integral part of the day-to-day operation of a system. In small systems, the same person may do all or part of each task. In large systems, individuals or whole teams may be required to perform a given function. A key document, frequently in use during this phase, is the Operations Manual (OM) detailing the tasks, activities and responsibilities. OM will require frequent updates to reflect the frequent streamlining of operational activities and tasks to ensure that the production environment is fully functional and performs as specified. The following is a checklist of systems operations key tasks and activities:

- Ensure that the systems and networks are running and available during the defined hours of Operations;
- Implement non-emergency requests during scheduled Outages, as prescribed in the OM document;
- Ensure that all processes, manual and automated, are documented in the operating procedures. These processes should comply with the system documentation;
- Acquire and store (stock) supplies (i.e. paper, toner, tapes, removable disk);
- Perform backups (day-to-day protection, contingency);
- Perform the physical security functions including ensuring adequate UPS, and that the personnel have proper security clearances and proper access privileges etc.;
- Ensure that the contingency planning for disaster recovery is current and tested;
- Ensure that the users are trained on current processes and new processes;
- Ensure that the service level objectives are kept accurate and are monitored;
- Maintain performance measurements, statistics, and system logs; examples of performance measures include volume and frequency of data to be processed in each mode, order and type of operations;
- Monitor the performance statistics, report the results and escalate problems when they occur;
- Monitor the effectiveness and adequacy of security controls.

8.2.6.1 Maintain Data/Software Administration

Data/Software Administration ensures that input and output data, as well as, the databases are up-to-date, correct and are continually checked for accuracy and completeness. This includes ensuring that automatic update patches are regularly scheduled, submitted and completed correctly. Software and databases should be maintained at (or near) the current maintenance level. The backup and recovery processes for databases are normally different from the periodic backups. The database backup and recovery should be performed as a Data/Software Administration task by a database administrator (DBA). A checklist of Data/Software Administration tasks and activities includes:

- Performing a periodic Verification/Validation of data and correcting the data related problems;
- Performing production control and quality control functions (Job submission, checking and corrections);
- Interfacing with other functional areas for daily checking/corrections;
- Installing, configuring, upgrading and maintaining data base(s). This includes updating processes, data flows, and objects (usually shown in diagrams);
- Developing and performing database backup and recovery routines for data integrity and recoverability (Ensure that these procedures are properly documented properly in the OM) ;
- Developing and maintaining a performance and tuning plan(s) for online process and data bases;
- Performing configuration/design audits to ensure that software, system, and parameters are correct.

8.2.6.2 Maintain System / Software

Daily system operations may necessitate that maintenance personnel identify the potential modifications needed to ensure that the system continues to operate as intended. Maintenance personnel may determine that updates to the system and databases are necessary to resolve errors or performance problems.

Modifications may be necessary to provide new capabilities or to take advantage of hardware upgrades or new releases of system and application software. New capabilities requirements may demand modified processes as described above. As a rule, all but the most incidental changes to the system will be in accordance with the FAS SDLC. Any change that requires additional funding authorization must go through the SDLC/CPIC processes to validate the business needs, assure funding, and update the investment portfolio.

8.2.7 Create Documentation and Deliverables

8.2.7.1 In-Process Review Report

The In-Process Review entails evaluating system performance, user satisfaction with the system, adaptability to changing business needs and new technologies that might improve the system. This review is diagnostic in nature and can trigger a project to re-enter a previous SDLC stage. The In-Process Review (IPR) occurs at predetermined milestones usually quarterly, but at least once a year. Ad hoc reviews may also occur as necessary. Document the results of any regular or ad hoc reviews in the IPR Report.

8.2.7.2 User Satisfaction Review Report

Consider employing User Satisfaction Reviews as a tool to determine the current user satisfaction with the performance capabilities of an existing application or initiate a proposal for a new system. Such input is useful as input to the IPR Report.

8.2.8 Hold Stage Review Activity

Review activities occur several times throughout this stage. At the completion of each In Process Review, a variety of determinations will be apparent, for example:

- The system is operating as intended and meeting performance expectations.
- The system is not operating as intended and needs corrections or modifications.
- The users are/are not satisfied with the operation and performance of the system.

The stage review should examine project process performance and the application of this SDLC for lessons learned and opportunities for improvement. Feedback with respect to application of the SDLC should be submitted to the FAS CIO.

9.0 DISPOSITION STAGE

9.1 Overview

9.1.1 Objectives

The Disposition Stage will be implemented to eliminate all, or in some cases major parts of, a system. This stage ends the life cycle process for the system.

In this stage, it is determined that the system is surplus and/or obsolete and eligible for shutdown. The emphasis of this stage is to ensure that data, procedures, and documentation are packaged and archived in an orderly fashion, making it possible to reinstall and bring the system back to an operational status, if necessary, and to retain all data records in accordance with GSA/FAS and Federal policies regarding retention of electronic records. The Disposition Stage represents the end of the systems life cycle. A Disposition Plan is prepared to address all facets of archiving, transferring, and disposing of the system and data. Particular emphasis on preservation of the data processed by the system so that it is effectively migrated to another system or archived in accordance with applicable records management regulations and policies for potential future access. The system disposition activities preserve information not only about the current production system but also about the evolution of the system through its life cycle.

9.1.2 Inputs

- Existing System
- Existing Deliverables and Work Products
- Disposition Request

9.1.3 High Level Activities

- I. Archive or Transfer Data
- II. Archive or Transfer Software Components
- III. Archive Life cycle Deliverables
- IV. End the System in an Orderly Manner
- V. Dispose of Equipment
- VI. Conduct Post-Termination Review
- VII. Create Documentation and Deliverables

9.1.4 Work Products

- Disposition Plan
- Post-Termination Review Report
- Archived System

9.2 Tasks and Activities

The objectives for all tasks identified in this stage are to retire the system, software, hardware and data. The tasks and activities actually performed are dependent on the nature of the project. The disposition activities ensure the orderly termination of the system and preserve vital information about the system so that some or all of it is available for reuse if necessary at some future date. Here again, put particular emphasis on proper preservation of the data processed by the system, so that the data are effectively migrated to another system or disposed of in accordance with applicable records management and program area regulations and policies for potential future access. These activities may be expanded, combined or deleted, depending on the size of the system.

9.2.1 Archive or Transfer Data

Transfer the data from the old system into the new system or, if it is obsolete, archive the data.

9.2.2 Archive or Transfer Software Components

Similar to data that transfer to the system or undergo archiving, migrate the software components to the new system or, if that is not feasible, dispose of the software accordingly.

9.2.3 Archive Life cycle Deliverables

Archive all system and SDLC documentation for future reference.

9.2.4 End the System in an Orderly Manner

Follow the Disposition Plan for the orderly breakdown of the system, its components and the data within.

9.2.5 Dispose of Equipment

Where possible, reusing hardware or software should comply with license or other agreements with the vendor and/or developer and with government regulations. There is rarely a need to destroy hardware, except for some storage media that contains sensitive information and that cannot be sanitized without destruction. In situations in which the storage media cannot be sanitized appropriately, removal and physical destruction of the media may be possible so that the remaining hardware may be reused elsewhere in the organization. If it is obsolete, notify the property management office to excess the hardware. Ensure all information systems media is properly disposed of in accordance with the Security Disposition Plan.

9.2.6 Conduct Post-Termination Review

The Project Lead will conduct this review at the end of the Disposition Stage and again within 6 months after disposition of the system.

9.2.7 Create Documentation and Deliverables

Complete the following work products during the Disposition Stage.

9.2.7.1 Disposition Plan

The objectives of the plan are to end the operation of the system in a planned, orderly manner and to ensure that system components and to properly archive or incorporate data into other systems. This includes removing the active support by the operations and maintenance organizations. The users will play an active role in the transition. All concerned groups informed of the progress and target dates. The decision to proceed with Disposition is an outcome of recommendations and approvals deriving from an IPR or from a timetable specified by the TRB or FAS Management Council. See Appendix C-23 for the Disposition Plan Outline.

This plan will includes a statement of the decision criteria for disposal, a description of any replacement system, and a list of tasks/activities (transition plan) with estimated dates of completion and the notification strategy. It also includes:

- The responsibilities for future residual support issues such as identifying media alternatives if technology changes
- New software product transition plans and alternative support issues (once the application is removed);
- Parallel operations of retiring and archiving of the software product
- Associated documents and movement of logs and code, and
- Accessibility of archive, data protection identification, and audit applicability.

9.2.7.2 Post-Termination Review Report

The PMO will produce a report at the end of the process that details the findings of the Disposition Stage review. It includes details of where to find all products and documentation that has been archived.

9.2.7.3 Archived System

This is the packaged set of data and documentation containing the archived application. Format is digital, under configuration control, and established for the specific project.

9.2.8 Hold Stage Review Activity

The Post-Termination Review is performed after the end of all other stage activities and within 6 months after disposition of the system. The Post-Termination Review Report documents the lessons learned from the shutdown and archiving of the terminated system.

The stage review should examine project process performance and the application of this SDLC for lessons learned and opportunities for improvement. Feedback with respect to application of the SDLC should be submitted to the FAS CIO.

10.0 SDLC TAILORING FOR INDIVIDUAL PROJECTS

An important objective of an SDLC methodology is to provide flexibility that allows tailoring of the methodology to suit the characteristics of a particular system development effort. One methodology does not fit all sizes and types of system development efforts. For instance, it is not reasonable to expect a very small system development project to produce 37 deliverables. Additionally, a different approach might be needed for a high-risk system development project that has very uncertain functional and technical requirements at the beginning of development.

The SDLC framework implements well-defined processes in a life cycle model that can be adapted to meet the specific requirements or constraints of any project. This section provides guidelines for adapting the life cycle processes to fit the characteristics of the project. These guidelines help ensure that there is a common basis across all projects for planning, implementing, tracking, and assuring the quality of the work products.

The SDLC framework has built-in flexibility. All of the stages and activities can be adapted to any size and scope project. The SDLC can be successfully applied to new development projects, operations and maintenance, and customization of commercial software.

The SDLC can be compressed to satisfy the needs of a small project, expanded to include additional activities or work products for a large or complex project, or supplemented to accommodate additional requirements. Any modifications to the life cycle should be consistent with the established activities, documentation, and quality standards included in the SDLC. Project teams are encouraged to tailor the SDLC as long as the fundamental SDLC objectives are retained and quality is not compromised.

The following are some examples of SDLC tailoring:

- Schedule stages and activities in concurrent or sequential order
- Repeat, merge, or eliminate stages, activities, or work products
- Include additional activities, tasks, or work products in a stage
- Change the sequence or implementation of activities
- Change the development schedule of the work products
- Combine or expand activities and the timing of their execution

The Project Manager should consider the size, complexity, and scope of the project when preparing SDLC documentation. There are a few essential tasks and work products that cannot be “tailored out” even when a COTS software product will be used:

- Detail project plans must be prepared including project schedule and project costs
- Detailed functional requirements must be defined
- A determination of the feasibility of the project and a look at possible alternatives including continuing with the status quo.
- The system must be in compliance with GSA/FAS policies and practices such as capital planning, enterprise architecture, and security
- The system must be adequately tested
- The users must be adequately trained
- Operations and maintenance documentation must be completed

The phases described in the SDLC do not intend to constrain systems development such that one phase must be completed before another phase begins. Many phases may be on-going at one time using approaches such as Rapid Application Development (RAD). Documents are identified according to a recommended creation, revision, and finalization phase in order to guide those responsible for development.

10.1 Requirements versus Guidelines

Any project life cycle that satisfies the information requirements represented by SDLC life cycle products and incorporates stage reviews satisfying both technical and investment governance expectations may be an acceptable tailored rendering of SDLC requirements. This SDLC document provides the framework elements from which a project life cycle must be tailored. The life cycle stage and products contained in this document represent guidelines for packaging life cycle activities and information in a standard, predictable way.

Across the core set of stage and products, the SDLC provides a set of Project Types that represent tailored interpretations of the SDLC. Each project type is an intermediate step between the core SDLC and a project life cycle, used for individual projects that conform to project type criteria.

The bottom line is that the FAS PM is responsible for defining a project life cycle that satisfies core SDLC mandates on information flow and stage review-based governance, satisfies GSA/FAS requirements, and provides a foundation for establishing management oversight of vendor work plans and development methodology.

The requirements of the project types embodied in the respective product matrices represent the SDLC mandate on individual projects. FAS OCIO must approve project level tailoring plans describing adaptation of this SDLC mandate. If a project proposes to take exception to fundamental SDLC disciplines relative to information/products, then the Exception Process must be utilized to solicit approval for those exceptions. FAS OCIO will be responsible for evaluating and transitioning commonly used tailoring and/or exceptions into the SDLC as additional “standards”, providing a means of keeping the SDLC current with FAS practice, eliminating the requirement to describe tailoring and/or exceptions for common work patterns.

A limited set of project types and project life cycle templates used across FAS projects is desirable because such an approach facilitates standardized project planning, management, and control processes. Review and approval of tailored project life cycles is an integral component of OCIO project governance because of the need to balance project needs for flexibility with enterprise need for predictability and repeatability.

10.2 Definition of a Project

A project is a unique venture with a beginning and an end, undertaken by people to meet established goals within defined constraints of time, resources, and quality. A project is defined regardless of its budget source.

In other words, it doesn't matter the budget source (e.g., DME, O&M, Steady State) – it is still a project.

10.3 Classification Schema

The FAS OCIO PMO has developed a classification schema based on the amount of Work Effort (measured in total Staff Hours) charged to a project.

Based on the Work Effort, a project will be classified into one of four classes, numbered one through four where a Level 1 project is small with a limited number of requirements and a Level 4 project is more complex and requires more oversight, approvals and deliverables.

The initial classification of a project can be increased by the inclusion of certain risk factors such as:

- Size of the project team
- Number of applications involved
- Familiarity of the technology to GSA (is it new technology?)
- Level of understanding of the requirements
- Political profile/impact

Depending on the calculated level of risk, these risk factors have the ability to increase a project's class. Reference Appendix C-27 for the Project Classification Schema. Table 10-1 defines the various artifacts

prescribed by the SDLC and the lifecycle in which it is created/updated/finalized, and whether it is required for the various project class. It is important to remember that tailoring is not about eliminating what must be done during the life cycle represented by SDLC stage. Tailoring is about repackaging information (product tailoring), rearranging activities (stage tailoring), and altering the formality and/or level of detail of activities and/or products.

Core Phases and Products Life Cycle Work Products	Concept Approval	Planning	Requirements Analysis	Design	Development/ Integration & Test	Implementation	Operations & Maintenance	Disposition	Project Classification Schema			
									1	2	3	4
Draft Business Case (DBC)	C	F	*	*	*	*	*					
Security Risk Assessment	C	R	F	*	*	*				X	X	X
Risk Management Plan (RMP)	C	C	R	R	F		*	*				X
Cost-Benefit Analysis (CBA)		C	R	R	F							
Feasibility Study		C	F									
Business Case (OMB Exhibit)		C/F	*	*	*	*	*	*			X	X
Concept of Operations (ConOps)		C	R	R	R	F	*	*				X
Project Management Plan (PMP)/PMP-Lite	C	C	R	R	F	*					X	X
Configuration Management Plan (CMP)		C	R	R	F	*	*					X
Quality Assurance Plan (QAP)		C	R	R	F	*	*					X
System Security Plan (SSP)		C	R	R	F	*	*			o	X	X
Acquisition Plan		C	R	R	F	*	*					
System Engineering Management Plan (SEMP)		C/F	*	*	*	*	*	*				
Functional Requirements Document (FRD)			C	F						o	X	X
Requirements Traceability Matrix (RTM)			C	R	F	*	*			X	X	X
Interface Control Document (ICD)			C	R	F	*	*			o	o	X
Privacy Act Notice/Privacy Impact Assessment		C	C	F		V				X	X	X
Test Plan (PT)		C	C	R	F	*	*			X	X	X
Conversion Plan				C	F	*						
System Design Document (SDD)				C	F		*			o	o	X
Implementation Plan (IMP)				C	F							X
Maintenance Manual (MM)				C	F	*	*			o	o	X
Operations Manual (OM) (System Administration Manual)				C	F	*	*			o	o	X
Training Plan (TP)				C	F	*	*			o	o	X
User Manual (UM)				C	F	*	*			o	o	X
IT Contingency Plan				C	F	*	*					
Software Development Document/Folder (SDF)					C/F	*	*					
System Software					C/F	*	*					
Test Files/Data					C/F		*					
Test Analysis Report (TAR)					P	*	*		X	X	X	X
Test Problem Report					P	P	P		X	X	X	X
User Acceptance Test Report					P				X	X	X	X
*IT Systems Security Certification & Accreditation					C/F					X	X	X
Security Vulnerability Scan					C/E	F	*		X	X	X	X
Delivered System						C/F	*					
Change Implementation Notice (CIN)						C/F	*					
Post-Implementation Review (PIR)						C/F	*					X
In-Process Review Report (IPR)							P					X

Core Phases and Products Life Cycle Work Products	Concept Approval	Planning	Requirements Analysis	Design	Development/ Integration & Test	Implementation	Operations & Maintenance	Disposition	Project Classification Schema			
									1	2	3	4
User Satisfaction Report							P		X	X	X	X
Disposition Plan								C/F				
Post-termination Review Report								P				
Archived System								C/F				

KEY: C=Create, E=Execute, F=Finalize, M=Monitor, P=Produce, R=Revise, V=Validate, *=Update if needed, lower case=optional

Table 10-1. Life Cycle Artifacts by Project Class

10.4 SDLC Tailoring Approval Process

The Office of Chief Information Officer (OCIO) Program Management Office (PMO) is responsible for SDLC oversight regarding the development of IT projects and their compliance within the SDLC framework. If a project proposes a project life cycle outside the tailoring guidelines presented in this document, it must be documented in the project's PMP and presented to the FAS PMO as appropriate. FAS PMO staff should be included in the project life cycle planning process; Program Sponsors, Program Managers, and Project Leads will need to coordinate with FAS PMO staff to discuss the proposed tailoring of the SDLC, initially during System Concept Development stage and, subsequently, during the Planning stage.

10.5 Life cycle Strategies

Software development methodologies aid in understanding the software development process. They assist planning by defining the expected sequence of activities, the products that flow between those activities, and management activities including reviews and milestones. The life cycle model is used as a communications tool among team members.

Project managers need a documented process and clear criteria to choose an appropriate life cycle model from among the numerous models that exist. These include waterfall, incremental, evolutionary, spiral, and agile development. Selection of an inappropriate life cycle can result in a system that does not satisfy user needs and increases costs and schedules.

The following sections provide descriptions of some development techniques that can be used with the SDLC.

10.5.1 Waterfall

The waterfall mode, as shown in figure 10-5, implies an ideal situation – the activities are performed once in the sequence indicated. The phases occur sequentially with the output on one phase providing input to the next phase. The waterfall model defines all requirements of the system, designs software to satisfy those requirements, develops the system based on the design, integrates, tests, and finally delivers a totally complete software system to a customer. Feedback and rework are allowed only to the previous step. While the waterfall model provides a structured, disciplined method for software development, it is a risky choice for new development because it inhibits flexibility. With a single pass through the process, most integration problems surface too late. Also, a completed product is not available until the end of the process which typically discourages user involvement.

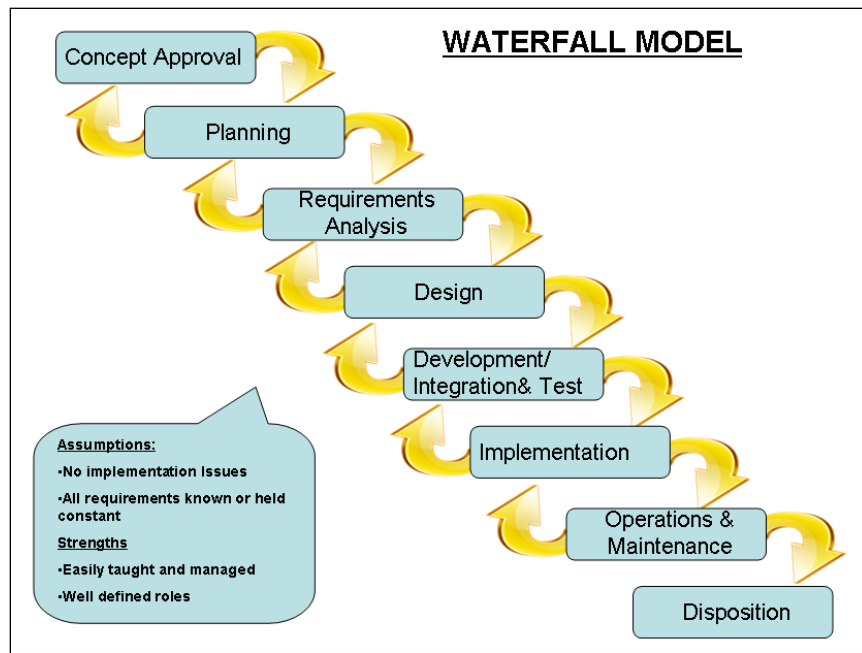


Figure 10-5. Waterfall Model

10.5.2 Evolutionary

The evolutionary model, as shown in figure 10-6, recognizes up front that all requirements needed to define the system are not fully capable of being defined or implemented, and then proceeds to develop the system as a series of builds. An initial core capability is developed to meet minimal essential requirements and is characterized by a flexible, modular structure. It also includes provisions for the evolutionary addition of future functionality and changes as requirements are further defined.

Each build is operational and delivered to the user for use and evaluation. This model assumes that the evaluation period will allow the user to define new requirements for the next build. Because each build is providing the user with some operational capability, resources must be allocated to the operations and maintenance activities that have to accompany each build.

Evolutionary models are particularly suited to situations where, although the general scope of the system is known, only a basic core of user functional characteristics can be defined or detailed systems requirements are difficult to quantify or articulate.

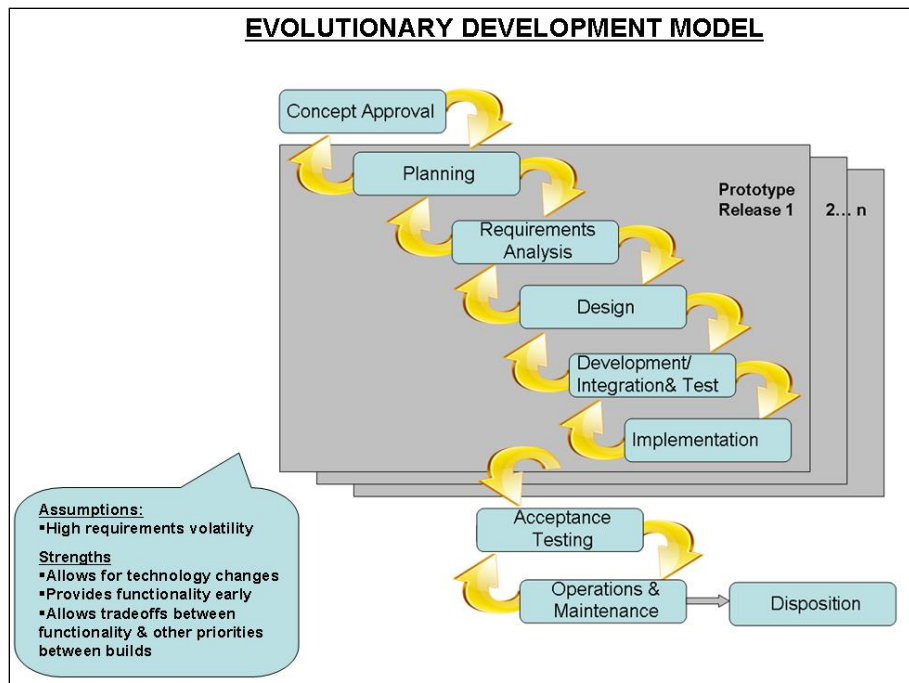


Figure 10-6. Evolutionary Development Model

10.5.3 Incremental

The incremental model, as shown in figure 10-7, involves developing the system in groups of functional capabilities. The system is partitioned into increments whose development is phased over the total development cycle. It allows users to employ part of the product before it is entirely completed.

The incremental model, similar to the waterfall model, determines user needs by defining all requirements, but then proceeds to design, develop, test, and deliver the system in a scheduled set of builds. Each successive build satisfies a subset of the overall requirements. This process is repeated until the entire product has been developed.

Incremental delivery models are characterized by a build-a-little, test-a-little approach to deliver an initial functional subset of the final capability. This subset is subsequently upgraded or augmented until the total scope of the stated user requirement is satisfied. An incremental model is most appropriate for low to medium risk projects, when user requirements can be fully defined, or assessment of other considerations (e.g., risks, funding, schedule, project size) indicate that a phased approach is the most prudent.

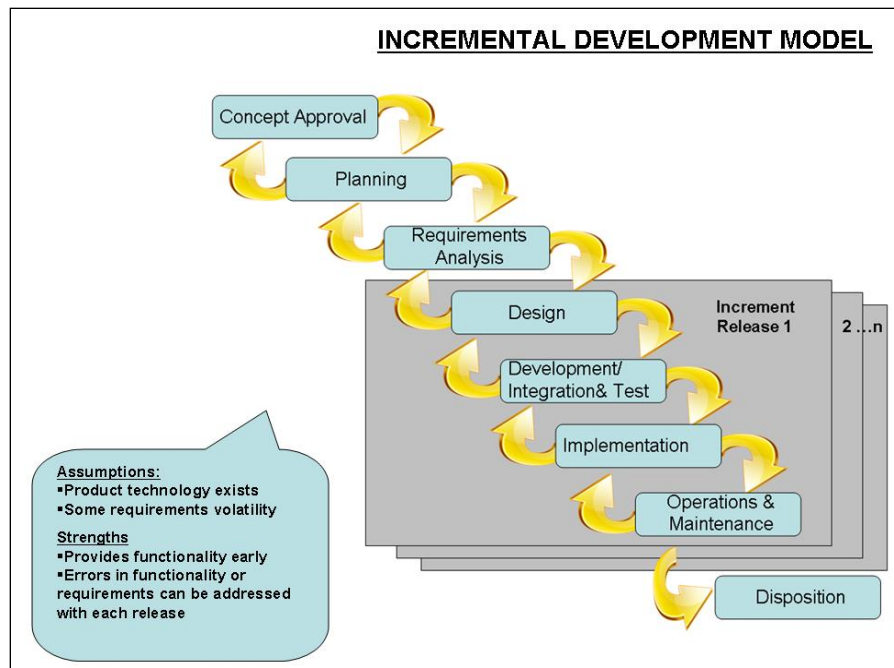


Figure 10-7. Incremental Development Model

10.5.4 Spiral

The spiral development, as shown in figure 10-8, model represents the activities related to software development as a spiraling progression of events that moves outward from a theoretical center. For each development phase from concept approval through design, this model places great emphasis on decision making to ensure management of all aspects of risks. Each cycle of the spiral includes defining the objectives and evaluating alternatives and constraints; evaluating the alternatives and their potential risks; developing and verifying the compliance of an interim product (e.g., prototype or document); and planning for the next phase – using knowledge gained from the previous phases.

Spiral development emphasizes evaluation of alternatives and risk assessment. These are addressed more thoroughly than with the other life cycle models. A review at the end of each phase ensures commitment to the next phase or identifies the need to rework a phase if necessary.

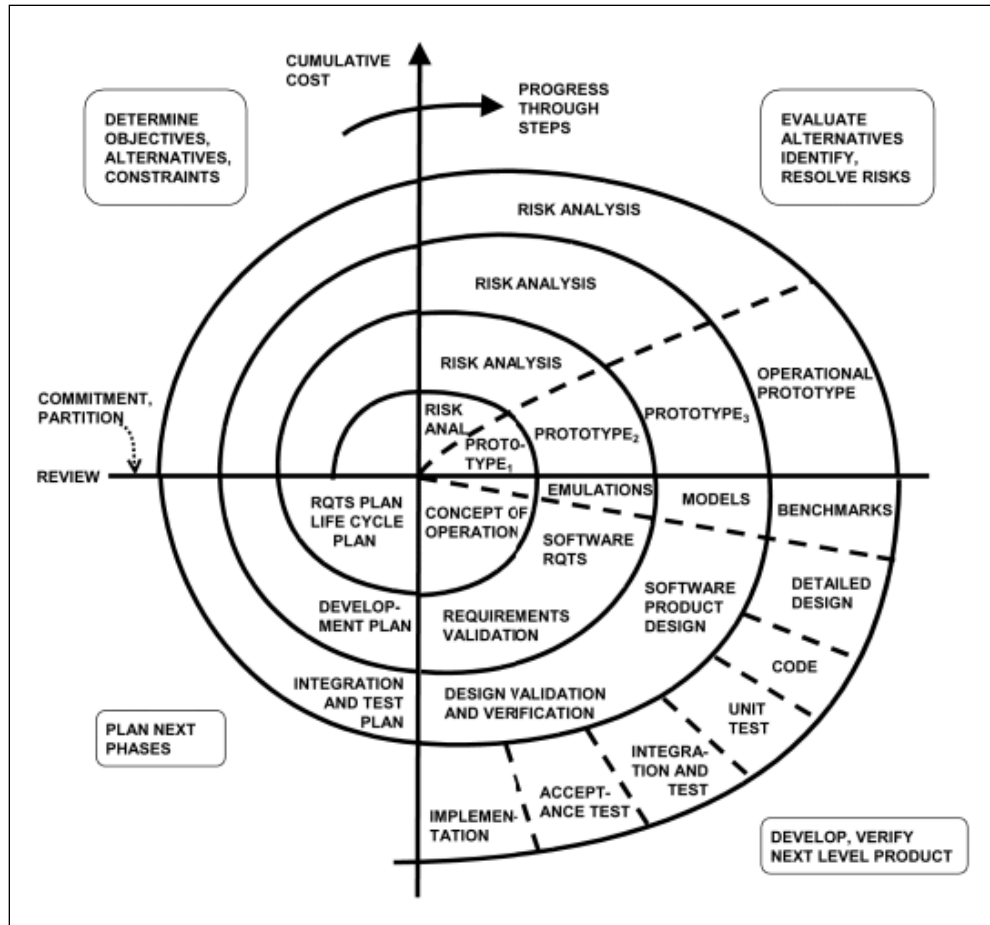


Figure 10-8. Spiral Development Model

10.5.5 Agile

Agile is an iterative and incremental mode of development wherein the entire development life cycle is broken down into small iterations. The project scope and requirements are defined during the planning phase at a very broad level. Also included in this definition is the number of planned iterations, duration, and scope of each iteration.

After this initial planning, the Agile approach consists of many rapid iterative planning and development cycles, allowing a project team to constantly evaluate the evolving product and obtain immediate feedback from users or stakeholders. The team learns and improves the product, as well as their working methods, from each successive iteration. At the beginning of each iteration, more detailed planning, requirements, design, build and test takes place within the duration allowed (typically 2-4 weeks) with a product produced at the end of the iteration. This approach allows for immediate modifications of the product as requirements come into view. For the best chances of success, an agile project requires a dedicated full-time project team that includes a customer or end user, where team members work from the same location.

Figure 10-9 depicts the Agile Development Model.

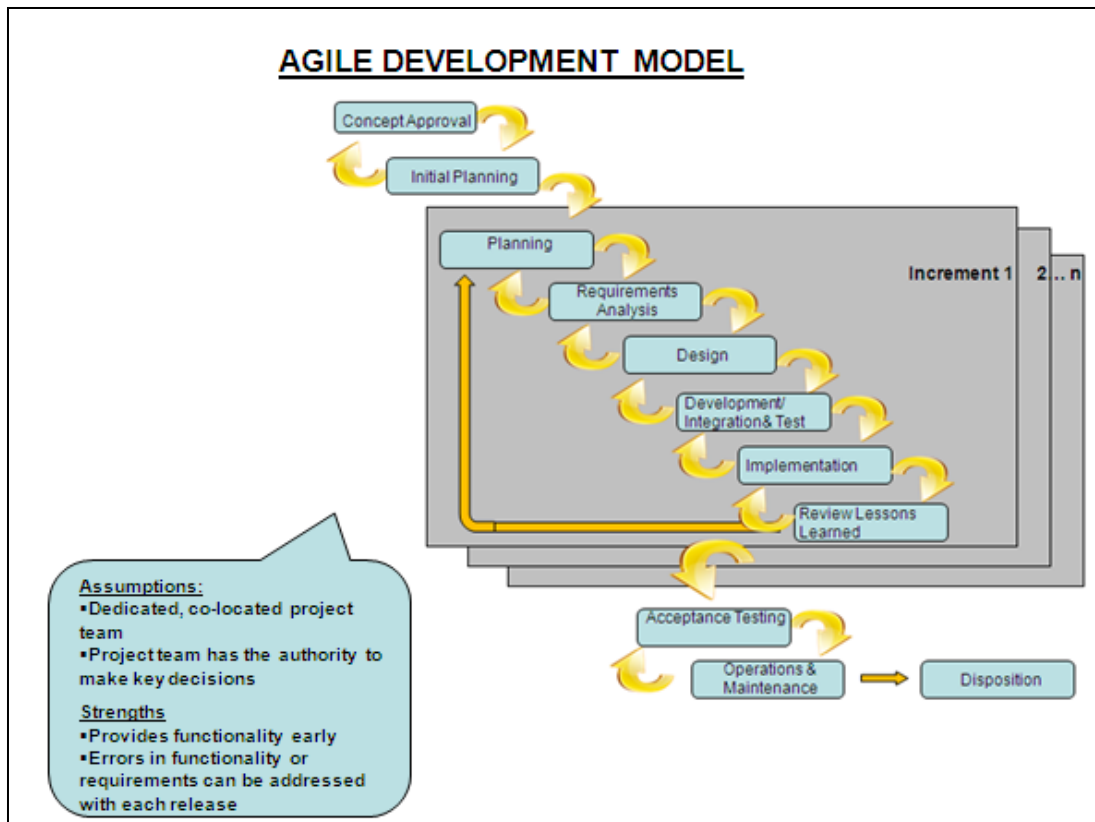


Figure 10-9. Agile Development Model

10.5.6 Advantages/Disadvantages of each Life Cycle Strategy

Each life cycle strategy offers its own advantages and disadvantages and special attention must be given to ensure the project selects the correct strategy – given the project's goals, objectives, risks, and budget constraints.

Figure 10-10 describes the advantages and disadvantages of each life cycle.

Life cycle Strategies	Criteria	Advantages	Disadvantages
Waterfall	Budget: High Time: Long-term Requirements: Stable	<ul style="list-style-type: none"> Clearly defined stages Assures delivery of initial requirements Well documented process and results 	<ul style="list-style-type: none"> Lack of measurable progress within stages Cannot accommodate changing requirements Resistant to time and budget constraints
Incremental	Budget: High Time: Short-term Requirements: Stable Or Budget: Low Time: Long-term Requirements: Stable	<ul style="list-style-type: none"> Supports early and periodic results Measurable progress Supports parallel development efforts 	<ul style="list-style-type: none"> Demands increased project management attention Can increase resource requirements Little support for changing requirements
Evolutionary	Budget: Low Time: Long-term Requirements: Dynamic	<ul style="list-style-type: none"> Supports changing requirements Minimizes time required to implement initial capability Achieves economy of scale for enhancements 	<ul style="list-style-type: none"> Increases project management complexity Initial capability only partially satisfies requirements and is not complete functionality Risk of not knowing when to end the project
Spiral	Budget: High Time: Long-term Requirements: Dynamic	<ul style="list-style-type: none"> Supports changing requirements Allows for extensive use of prototypes More accurately captures requirements 	<ul style="list-style-type: none"> Increases project management complexity Defers production capability to end of the SDLC Risk of not knowing when to end the project
Agile	Budget: Medium Time: Short-term Requirements: Dynamic	<ul style="list-style-type: none"> Supports changing requirements and priorities Supports early and predictable results Incorporates lessons learned and improvements throughout the project's lifecycle 	<ul style="list-style-type: none"> Increases project management complexity Requires dedicated project team Initial capability only partially satisfies requirements and is not complete functionality Risk of not knowing when to end the project

Figure 10-10. Alternative Life cycle Strategies

APPENDIX A: GLOSSARY

-A-

Acceptance Test - Formal testing conducted to determine whether or not a system, subsystem, or configuration item satisfies its acceptance criteria and to enable the customer to determine whether or not to accept. See User Acceptance Test.

Accreditation - Formal declaration by an accrediting authority that a computer system is approved to operate in a particular security mode using a prescribed set of safeguards.

Acquisition Plan - A formal document showing how all hardware, software, and telecommunications capabilities, along with resources, are to be obtained during the life of the project.

Adaptability - The ease with which software satisfies differing system constraints and user needs.

Allocated Baseline - The approved documentation that describes the design of the functional and interface characteristics that are allocated from a higher-level configuration item. See Baseline.

Availability - The degree to which a system (or system component) is operational and accessible when required for use.

-B-

Baseline - A work product (such as software or documentation) that has been formally reviewed, approved, and delivered and can only be changed through formal change control procedures. See Allocated Baseline, Functional Baseline, Operational Baseline, and Product Baseline.

Business Process Reengineering - The discipline of redesigning organizational business processes, cultural perspectives, and, frequently, technology deployment, to achieve significant improvements in costs, time, service, and quality.

-C-

Capability - A measure of the expected use of a system.

Capacity - A measure of the amount of input a system could process and/or amount of work a system can perform; for example, number of users, number of reports to be generated.

Certification - Comprehensive analysis of the technical and non-technical security features and other safeguards of a system to establish the extent to which a particular system meets a set of specified security requirements.

Change - In Configuration Management, a formally recognized revision to a specified and documented requirement. See Change Control, Change Directive, Change Impact Assessment, Change Implementation Notice.

Change Control - In Configuration Management, the process by which a change is proposed, evaluated, approved (or disapproved), scheduled, and tracked. See Change, Change Directive, Change Impact Assessment, Change Implementation Notice.

Change Control Documents - Formal documents used in the configuration management process to track, control, and manage the change of configuration items over the systems development or maintenance life cycle. See System Change Request, Change Impact Assessment, Change Directive, and Change Implementation Notice.

Change Directive - The formal Change Control Document used to implement an approved change. See Change Control Documents.

Change Impact Assessment - The formal Change Control Document used to determine the effect of a proposed change before a decision is made to implement it. See Change Control Documents.

Change Implementation Notice - The formal Change Control Document used to report the actual implementation of a change in a system. See Change Control Documents.

Computer System Security Officer - The person who ensures that all Computer and Telecommunications Security (C&TS) activities are undertaken at the user site. These typically include security activities for planning; awareness training; risk management; configuration management; certification and accreditation; compliance assurance; incident reporting; and guidance and procedures.

Concept of Operations - A formal document that describes the user's environment and process relative to a new or modified system; defines the users, if not already known; also known as a CONOPS.

Configuration - The functional and/or physical collection of hardware and software components as set forth in formal documentation. This is the aggregate of requirements, design, and implementation that define a particular version of a system (or system component). See Configuration Control, Configuration Item, Configuration Management, Configuration Management Plan, Configuration Status Accounting.

Configuration Audit - This is a formal review of a project for the purpose of assessing compliance with the Configuration Management Plan.

Configuration Control - This is the process of evaluating, approving or (disapproving), and coordinating changes to hardware/software configuration items.

Configuration Control Board - The formal entity charged with the responsibility of evaluating, approving (or disapproving), and coordinating changes to hardware/software configuration items.

Configuration Item - An aggregation of hardware and/or software that satisfies an end-user function; treated as a single entity in the configuration management process. This is a component of a system requiring continuous control over the course of development and throughout the life cycle of the system.

Configuration Management - The discipline of identifying the configuration of a hardware/software system at each life cycle stage for the purpose of controlling changes to the configuration and maintaining the integrity and traceability of the configuration through the entire life cycle.

Configuration Management Plan - A formal document that establishes formal configuration management practices in a systems development/maintenance project. See Configuration Management.

Configuration Status Accounting - The recording and reporting of the information that is needed to effectively manage a configuration; including a listing of the approved configuration identification, status of proposed changes to the configuration, and the implementation status of approved changes. See Configuration.

Contingency Plan - A formal document that establishes continuity of operations processes in case of a disaster. Includes names of responsible parties to be contacted, data to be restored, and location of such data.

Conversion Plan - A formal document that describes the strategies involved in converting data from an existing system to another hardware or software environment.

Corrective Maintenance - Maintenance performed to correct faults in hardware or software.

Cost-Benefit Analysis - The comparison of alternative courses of action, or alternative technical solutions, for the purpose of determining which alternative would realize the greatest cost benefit; cost-benefit analysis is also used to determine if the system development or maintenance costs still yield a benefit or if the effort should stop.

Critical Path - Used in project planning; the sequence of activities (or tasks) that must be completed on time to keep the entire project on schedule; therefore, the time to complete a project is the sum of the time to complete the activities on the critical path.

-D-

Data Dictionary - A repository of information about data, such as its meaning, relationships to other data, origin, usage and format. A data dictionary manages data categories such as aliases, data elements, data records, data structure, data store, data models, data flows, data relationships, processes, functions, dynamics, size, frequency, resource consumption and other user-defined attributes.

Design Stage - The period of time in the systems development life cycle during which the designs for architecture, software components, interfaces, and data are created, documented, and verified to satisfy system requirements.

Development/Integration & Test Stage - The period of time in the systems development life cycle to convert the work products of the Design Stage into a complete system. Subsystem integration, system, security, and user acceptance testing are conducted; done prior to the Implementation Stage.

Disposition Stage - The time when a system has been declared surplus and/or obsolete and the task performed is either eliminated or transferred to other systems.

Disposition Plan - A formal plan providing the full set of procedures necessary to end the operation of the system in a planned, orderly manner and to ensure that system components and data are properly archived or incorporated into other systems.

Draft Business Case - A formal document created during the System Concept Development Stage, which lists the business case for initiating the system or project. It contains responsible persons, projected costs associated with the investment, risks, assumptions, scope, schedule, milestones, etc.

-E-

Entity - Represents persons, places, events, things, or abstractions that are relevant to the FAS and about which data are collected and maintained.

-F-

Fault Tolerance - The ability of a system (or system component) to continue normal operation despite the presence of hardware or software faults.

Feasibility Study - A formal study to determine the feasibility of a proposed system (new or enhanced) in order to make a recommendation to proceed or to propose alternative solutions.

Field Test - Testing that is performed at the user site.

Functional Baseline - The approved documentation that describes the functional characteristics of the system, subsystem, or component. See Baseline.

Functional Configuration Audit - An audit to ensure that the delivered configuration item has met the functional requirements. See Audit.

Functional Requirement - A requirement that specifies a function (activity or behavior, based on a business requirement) that the system (or system component) must be capable of performing.

Functional Requirements Document - A formal document of the business (functional) requirements of a system; the baseline for system validation.

Functional Test - Testing that ignores the internal mechanism of a system (or system component) and focuses solely on the outputs generated in response to selected inputs and execution conditions. Same as black box testing.

-G-

Gantt Chart - A list of activities plotted against time, showing start time, duration, and end time; also known as a bar chart.

-I-

Implementation - Installing and testing the final system, usually at the user (field) site; the process of installing the system.

Implementation Stage - The period of time in the systems development life cycle when the system is installed, made operational, and turned over to the user (for the beginning of the Operations and Maintenance Stage).

Implementation Plan - A formal document that describes how the system will be installed and made operational.

In-Process Review - Formal review conducted (usually annually) during the Operations and Maintenance Stage to evaluate system performance, user satisfaction with the system, adaptability to changing business needs, and new technologies that might improve the system.

In-Process Review Report - A formal document detailing the findings of the In-Process Review. See In-Process Review.

Inspection - A semiformal-to-formal technique in which software requirements, design, or code are examined in detail by a person or group other than the originator to detect errors. See Peer Review, Walk-through.

Integrated Product Team - A multidisciplinary group of people who support the Project Lead in the planning, execution, delivery and implementation of life cycle decisions for the project.

Integration Test - Testing in which software components, hardware components, or both are combined and tested to evaluate the interaction between them.

Integrity - The degree to which a system (or system component) prevents unauthorized access to, or modification of, computer programs or data.

Iterative - A procedure in which repetition of a sequence of activities yields results successively close to the desired state; for example, an iterative life cycle in which two or more stage are repeated until the desired product is developed.

Interface Control Document - Specifies the interface between a system and an external system(s).

Interoperability - A measure of the ability of two or more systems (or system components) to exchange information and use the information that has been exchanged. Same as Compatibility.

Information Technology Systems Security Certification and Accreditation - A formal set of documents showing that the installed security safeguards for a system are adequate and work effectively.

-M-

Maintainability - The ease with which a software system (or system component) can be modified to correct faults, improve performance or other attributes, or adapt to a changed environment.

Maintenance Manual - A formal document that provides systems maintenance personnel with the information necessary to maintain the system effectively.

Methodology - A set of methods, procedures, and standards that define the approach for completing a system development or maintenance project.

Milestone - In project management, a scheduled event that is used to measure progress against a project schedule and budget.

-N-

Non-technical - Relating to agreements, conditions, and/or requirements affecting the management activities of a project. Compare to Technical.

-O-

Operational Baseline - Identifies the system accepted by the users in the operational environment after a period of onsite test using production data. See Baseline.

Operations Manual - A formal document that provides a detailed operational description of the system and its interfaces.

Operations and Maintenance (O&M) Stage - The period of time in the systems development life cycle during which a software product is employed in its operational environment, monitored for satisfactory performance, and modified as necessary to correct problems or to respond to changing requirements.

-P-

Peer Review - A formal review where a person or group other than the originator examines a product in detail. See Inspection, Walk-through.

Perfective Maintenance - Software maintenance performed to improve the performance, maintainability, or other attributes of a computer program.

Performance Measures - A category of quality measures that address how well a system functions.

Performance Measurement and Capacity Planning - A set of procedures to measure and manage the capacity and performance of information systems equipment and software.

Performance Review - Formal review conducted to evaluate the compliance of a system or component with specified performance requirements.

Physical Configuration Audit - An audit to ensure that the configuration item has met all physical attributes listed in the design requirements being delivered.

Pilot - An alternative work pattern to develop a system to demonstrate that the concept is feasible in an operational environment. Pilots are used to provide feedback to refine the final version of the product and are fielded for a preset, limited period of time. Compare to a Prototype.

Planning Stage - The period of time in the systems development life cycle in which a comprehensive plan for the recommended approach to the systems development or maintenance project is created. Follows the Concept Approval Stage, in which the recommended approach is selected.

Post-Implementation Review - A formal review to evaluate the effectiveness of the systems development effort after the system is operational (usually for at least six months).

Post-Implementation Review Report - A formal document detailing the findings of the Post-Implementation Review. See Post-Implementation Review.

Post-Termination Review - A formal review to evaluate the effectiveness of a system disposition.

Post-Termination Review Report - A formal document detailing the findings of the Post-Termination Review. See Post-Termination Review.

Privacy Act Notice - For any system that has been determined to be an official System of Records (in terms of the criteria established by the Privacy Act), a special notice must be published in the Federal Register that identifies the purpose of the system; describes its routine use and what types of information and data are contained in its records; describes where and how the records are located; and identifies who the System Manager is.

Procedure - A series of steps (or instructions) required to perform an activity. Defines "how" to perform an activity. Compare to Process.

Process - A finite series of activities as defined by its inputs, outputs, controls (for example, policy and standards), and resources needed to complete the activity. Defines "what" needs to be done. Compare to Procedure.

Process Model - A graphical representation of a process.

Process Review - A formal review of the effectiveness of a process.

Product Baseline - The set of completed and accepted system components and the corresponding documentation that identifies these products. See Baseline.

Production - A fully documented system, built according to the SDLC, fully tested, with full functionality, accompanied by training and training materials, and with no restrictions on its distribution or duration of use.

Product Review - A formal review of a product software (or document) to determine if it meets its requirements. Can be conducted as a peer review.

Project - The complete set of activities associated with all life cycle stage needed to complete a systems development or maintenance effort from start to finish (may include hardware, software, and other components); the collective name for this set of activities. Typically a project has its own funding, cost accounting, and delivery schedule.

Project Management Plan - A formal document detailing the project scope, activities, schedule, resources, and security issues. The Project Management Plan is created during the Planning Stage and updated through the Disposition Stage.

Prototype - A system development methodology to evaluate the design, performance, and production potential of a system concept (it is not required to exhibit all the properties of the final system). Prototypes are installed in a laboratory setting and not in the field, nor are they available for operational use. Prototypes are maintained only long enough to establish feasibility. Compare to a Pilot.

-Q-

Quality - The degree to which a system, component, product, or process meets specified requirements.

Quality Assurance - A discipline used by project management to objectively monitor, control, and gain visibility into the development or maintenance process.

Quality Assurance Plan - A formal plan to ensure that delivered products satisfy contractual agreements, meet or exceed quality standards, and comply with approved systems development or maintenance processes.

Quality Assurance Review - A formal review to ensure that the appropriate Quality Assurance activities have been successfully completed, held when a system is ready for implementation.

-R-

Records Disposition Schedule - Federal regulations require that all records no longer needed for the conduct of the regular business of the agency be disposed of, retired, or preserved in a manner consistent with official Records Disposition Schedules.

Records Management - The formal set of system records (for example, files, data) that must be retained during the Disposition Stage; the plan for collecting and storing these records.

Recoverability - The ability of a software system to continue operating despite the presence of errors.

Regression Test - In software maintenance, the rerunning of test cases that previously executed correctly in order to detect errors introduced by the maintenance activity.

Reliability - The ability of a system (or system component) to perform its required functions under stated conditions for a specified period of time.

Requirement - A capability needed by a user; a condition or capability that must be met or possessed by a system (or system component) to satisfy a contract, standard, specification, or other formally imposed documents.

Requirements Analysis Stage - The period of time in the systems development life cycle during which the requirements for a software product are formally defined, documented and analyzed.

Requirements Management - Establishes and controls the scope of system development efforts and facilitates a common understanding of system capabilities between the System Proponent, developers, and future users.

Requirements Traceability Matrix - Provides a method for tracking the functional requirements and their implementation through the development process.

Resource - In management, the time, staff, capital and money available to perform a service or build a product; also, an asset needed by a process step to be performed.

Review - A formal process at which an activity or product (for example, code, document) is presented for comment and approval; reviews are conducted for different purposes, such as peer reviews, user

reviews, management reviews (usually for approval) or done at a specific milestone, such as stage reviews (usually to report progress).

Review Report - A formal document that records the results of a review.

Risk - A potential occurrence that would be detrimental to the project; risk is both the likelihood of the occurrence and the consequence of the occurrence.

Risk Assessment - The process of identifying areas of risk; the probability of the risk occurring, and the seriousness of its occurrence; also called risk analysis.

Risk Management - The integration of risk assessment and risk reduction in order to optimize the probability of success (that is, minimize the risk).

Risk Management Plan - A formal document that identifies project risks and specifies the plans to reduce these risks.

-S-

Security - The establishment and application of safeguards to protect data, software, and hardware from accidental or malicious modification, destruction, or disclosure.

Security Risk Assessment - Tool that permits developers to make informed decisions relating to the acceptance of identified risk exposure levels or implementation of cost-effective measures to reduce those risks. See Requirements Analysis Stage.

Security Test - A formal test performed on an operational system, based on the results of the security risk assessment in order to evaluate compliance with security and data integrity guidelines, and address security backup, recovery, and audit trails. Also called Security Testing and Evaluation (ST&E).

Sensitive System - A system or subsystem that requires an IT Systems Security Certification and Accreditation; contains data requiring security safeguards.

Sensitivity Analysis - Assesses the potential effect on inputs (costs) and outcomes (benefits) that depends on the relative magnitude of change in certain factors or assumptions.

Software Development Document - Contains all of the information pertaining to the development of each unit or module, including the test cases, software, test results, approvals, and any other items that will help explain the functionality of the software.

Stage - A defined stage in the systems development life cycle; there are eight stages in the full, sequential life cycle.

Stage Review - A formal review conducted during a life cycle stage; usually at the end of the stage or at the completion of a significant activity.

System Change Request - The formal Change Control Document procedure used to request a change to a system baseline, provide information concerning the requested change, and act as the documented approval mechanism for the change. See Change Control Documents.

System Design Document - A formal document that describes the system architecture, file and database design, interfaces, and detailed hardware/software design; used as the baseline for system development.

Systems of Records Notice - Notice that is required to be published for any system that has been determined to be an official System of Records (in terms of the criteria established by the Privacy Act).

System Proponent - The organization benefiting from or requesting the project; frequently thought of as the "customer" for that project.

Systems Administration Manual - A manual that serves the purpose of an Operations Manual in a distributed (client/server) application. See Operations Manual, Client/Server.

Systems Analysis - In systems development, the process of studying and understanding the requirements (customer needs) for a system in order to develop a feasible design.

Systems Development Life cycle - A formal model of a hardware/software project that depicts the relationship among activities, products, reviews, approvals, and resources. Also, the period of time that begins when a need is identified (Concept Approval) and ends when the system is no longer available for use (disposition).

System Security Plan - A formal document that establishes the processes and procedures for identifying all areas where security could be compromised within the system (or subsystem).

System Test - The process of testing an integrated hardware/software system to verify that the system meets its documented requirements.

-T-

Test Case - A specific set of test data and associated procedures developed for a particular test.

Test Files/Data - Files/data developed for the purpose of executing a test; becomes part of a test case. See Test Case.

Testability - A metric used to measure the characteristics of a requirement that enable it to be verified during a test.

Test Analysis Report - Formal documentation of the software testing as defined in the Test Plan.

Test and Evaluation (T&E) - T&E occurs during all major stages of the development life cycle, beginning with system planning and continuing through the operations and maintenance stage, ensures standardized identification, refinement, and traceability of the requirements as such requirements are allocated to the system components.

Test and Evaluation Master Plan - The formal document that identifies the tasks and activities so the entire system can be adequately tested to assure a successful implementation.

Test Problem Report - Formal documentation of problems encountered during testing; the form is attached to the Test Analysis Report. See Test Analysis Report.

Test Readiness Review - A formal stage review to determine that the test procedures are complete and to ensure that the system is ready for formal testing.

Traceability - In requirements management, the identification and documentation of the derivation path (upward) and allocation path (downward) of requirements in the hierarchy.

Training Plan - A formal document that outlines the objectives, needs, strategy, and curriculum to be addressed for training users of the new or enhanced system.

-U-

Unit - the smallest logical entity specified in the design of a software system; must be of sufficient detail to allow the code to be developed and tested independent of other units. See Module.

Unit Test - In testing, the process of ensuring that the software unit executes as intended; usually performed by the developer.

Usability - The capability of the software product to be understood, learned, used and be of value to the user, when used under specified conditions.

User Acceptance Test - Formal testing conducted to determine whether or not a system satisfies its acceptance criteria and to enable the user to determine whether or not to accept the system. See Acceptance Test.

User Manual - A formal document that contains all essential information for the user to make full use of the new or upgraded system.

User Satisfaction Review - A formal survey used to gather the data needed to analyze current user satisfaction with the performance capabilities of an existing system or application; administered annually, or as needed.

-V-

Validation - The process of determining the correctness of the final product, system, or system component with respect to the user's requirements. Answers the question, "Am I building the right product?" Compare to Verification.

Verifiability - A measure of the relative effort to verify a requirement; a requirement is verifiable only if there is a finite cost-effective process to determine that the software product or system meets the requirement.

Verification - The process of determining whether the products of a life cycle stage fulfill the requirements established during the previous stage; answers the question, "Am I building the product right?" Compare to Validation.

Version - An initial release or re-release of a computer software configuration item, associated with a complete compilation or recompilation of the computer software configuration item; sometimes called a build. See Build.

Volatility - In requirements management, volatility is the degree to which requirements are expected to change throughout the systems development life cycle; opposite of stability.

-W-

Walk-through - A software inspection process, conducted by peers of the software developer, to evaluate a software component. See Inspection, Peer Review.

Work Breakdown Structure - In project management, a hierarchical representation of the activities associated with developing a product or executing a process; a list of tasks; often used to develop a Gantt chart.

Work Pattern - The complete set of life cycle stages, activities, work products, and reviews required to develop or maintain a software product or system; a formal approach to systems development.

APPENDIX B: ACRONYMS

BPR	Business Process Reengineering
C&A	Certification & Accreditation
CBA	Cost-Benefit Analysis
CCB	Change Control Board
CI	Configuration Item
CIN	Change Implementation Notice
CIO	Chief Information Officer
CM	Configuration Management
CONOPS	Concept of Operations
COTS	Commercial Off-The-Shelf
CPIC	Capital Planning and Investment Control
CRUD	Create, Read, Update, Delete
CSA	Configuration Status Accounting
DBA	Database Administrator
DBC	Draft Business Case
DBMS	Database Management System
DME	Development, Maintenance, and Enhancements
EA	Enterprise Architecture
EVM	Earned Value Management
EVMS	Earned Value Management System
FAR	Federal Acquisition Regulations
FAS	Federal Acquisition Service
FCA	Functional Configuration Audit
FIPS	Federal Information Processing Standards
FOIA/PA	Freedom of Information Act/Privacy Act
FRD	Functional Requirements Document
GAO	General Accounting Office
GPRA	Government Performance and Results Act
GSA	General Services Administration
ICD	Interface Control Document
IEEE/EIA	Institute of Electrical and Electronics Engineers/Electronic Industries Assoc.
IPR	In-Process Review
IPT	Integrated Product Team
IRB	Investment Review Board
ISO	International Standard Organization
IT	Information Technology

KDP	Key Decision Point
LAN	Local Area Network
LoB	Line of Business
NIST	National Institute for Standards and Technology
OCIO	Office of Chief Information Officer
O&M	Operations and Maintenance
OM	Operations Manual
OMB	Office of Management and Budget
OPS	Operations-
PCA	Physical Configuration Audit
PDR	Preliminary Design Review
PIA	Privacy Impact Assessment
PIR	Post-Implementation Review
PM	Project or Program Manager
PMBOK	Project Management Body of Knowledge
PMI	Project Management Institute
PMO	Program Management Office
PRRA	Paperwork Reduction Reauthorization Act
QA	Quality Assurance
RM	Risk Management
RTM	Requirements Traceability Matrix
SAM	Systems Administration Manual
SCR	System Change Request
SDLC	Systems Development Life cycle
SEMP	Systems Engineering Management Plan
SOW	Statement of Work
SSO	Services and Staff Offices
TP	Training Plan
TPR	Test Problem Report
WBS	Work Breakdown Structure

APPENDIX C: TEMPLATES

Appendix C is a compilation of the following individually stored templates:

C-1	Draft Business Case (DBC)
C-2	Cost-Benefit Analysis (CBA)
C-3	Feasibility Study
C-4	Risk Management Plan
C-5	Configuration Management Plan
C-6	Quality Assurance Plan
C-7	Concept of Operations
C-8	System Engineering Management Plan
C-9	Functional Requirements Document (FRD)
C-10	Test Plan (PT)
C-11	Interface Control Document (ICD)
C-12	Conversion Plan
C-13	System Design Document (SDD)
C-14	Implementation Plan (IP)
C-15	Maintenance Manual (MM)
C-16	Operations Manual (OM)
C-17	System Administration Manual (SAM)
C-18	Training Plan (TP)
C-19	User Manual (UM)
C-20	Contingency Plan (CP)
C-21	Test Analysis Report
C-22	Post-Implementation Review
C-23	Disposition Plan
C-24	Project Management Plan (PMP)
C-25	Project Management Plan – Lite
C-26	User Acceptance Report (UAR)
C-27	Project Classification Schema
C-28	Requirements Traceability Matrix (RTM)
External	OMB Exhibit(s) as defined by the OMB
External	Privacy Act Notice/Privacy Impact Assessment
External	IT Systems Security Certification & Accreditation
External	Security Risk Assessment
External	System Security Plan
External	IT System Security Certification and Accreditation
Project	System Software as designated by the project

Project	Test Files/Data as designated by the project
Project	Delivered System as designated by the project
Project	Archived System as designated by the project
Project	Test Problem Report as designated by the project
Project	Change Implementation Notice as designated by the project
Project	In-Progress Review as designated by the project
Project	Software Development Folder as designated by the project
Project	User Satisfaction Report